COUNTERING THE EFFECTS OF MEASUREMENT NOISE DURING THE IDENTIFICATION OF DYNAMICAL SYSTEMS

DISSERTATION

Odell R. Reynolds
Captain, USAF

AFIT/DS/ENG/96-13

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/DS/ENG/96-13

COUNTERING THE EFFECTS OF MEASUREMENT NOISE DURING THE IDENTIFICATION OF DYNAMICAL SYSTEMS

DISSERTATION

Odell R. Reynolds
Captain, USAF

AFIT/DS/ENG/96-13

The views expressed in this dissertation are those of the author and do not reflect the official policy or position of the Department of Defense or the U. S. Government.

AFIT/DS/ENG/96-13

COUNTERING THE EFFECTS OF MEASUREMENT NOISE DURING THE IDENTIFICATION OF DYNAMICAL SYSTEMS

DISSERTATION

Presented to the Faculty of the Graduate School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Doctor of Philosophy

Odell R. Reynolds, B.S. Electrical and Electronics Engineering, M.S.E.E.
Captain, USAF

December 1996

Approved for public release; Distribution Unlimited

AFIT/DS/ENG/96-13

COUNTERING THE EFFECTS OF MEASUREMENT NOISE DURING THE IDENTIFICATION OF DYNAMICAL SYSTEMS

Odell R. Reynolds, B.S. Electrical and Electronics Engineering, M.S.E.E.

Captain, USAF

Approved:

	Date
M. Pachter Meir Pachter, Chairman	11/26/96
Constantine H. Houpis Constantine H. Houpis	11/26/96
Mark E. Oxley Mark E. Oxley	11/26/96
Christopher D. Hall Christopher D. Hall, Dean's Representative	12/3/96

Accepted:

Robert A. Calico, Jr
Dean, School of Engineering

Acknowledgements

Firstly, thanks must be extended to the members of my committee whose suggestions helped bring this document together. Special thanks to my advisor, Dr. Meir Pachter, whose breadth of knowledge helped me keep an open mind. I hope he didn't tire of my endless questions. Secondly, thanks to my family, friends, and cellmates in the world of academia. They provided me with enough distractions to keep the process bearable. Finally, thanks to my wife Stacey, who stayed married to me through three plus years of AFIT education. Her endless support was a constant light at the end of the tunnel.

Odell R. Reynolds

Table of Contents

Page

List of Figures

Figure Page

Figure Page

List of Tables

List of Abbreviations

Notation

Scalars, Vectors, Matrices

Scalars are denoted by lower case letters in italic type. Vectors are denoted by lower case letters in boldface type. The n-dimensional vector $\mathbf{x}$ is made up of components x_i for $i = 1, \ldots, n$. Matrices are denoted by upper case letters in boldface type, as in the matrix $\mathbf{H}$, made up of elements H_{ij} (ith row, jth column).

Miscellaneous Notation

$\widehat{(\bullet)}$ - estimated value

$(\bullet)^T$ - transpose

$E\{\bullet\}$ - expected value

$\|\bullet\|$ - norm

$\mathcal{N}(a, b)$ - normally distributed with mean a and covariance b

$\mathcal{R}^x$ - x dimensional space of all real numbers

Common Symbols

a_i	: ith denominator coefficient
b_i	: ith numerator coefficient
A_k	: real component of kth frequency response measurement
B_k	: imaginary component of kth frequency response measurement
$\mathbf{H}$	: regression matrix
m	: number of measurements
M_k	: magnitude of kth frequency response measurement
M_{kdB}	: magnitude, in decibels, of kth frequency response measurement
n	: order of the system

p	: factor of ten for increasing the original noise level
$\mathbf{R}$	: linear regression equation error covariance (noise covariance) matrix
	$(\bullet)_k$ - on kth frequency response measurement
	$(\bullet)_{AB}$ - for phasor measurements
	$(\bullet)_{dB^\circ}$ - for dB magnitude and degree phase measurements
	$(\bullet)_{M\phi}$ - for magnitude and phase measurements
$\mathbf{T}$	: transformation matrix
	$(\bullet)_k$ - from phasor noise to equation error
	$(\bullet)_{dB^\circ}$ - from magnitude/phase noise to phasor noise
	$(\bullet)_{M\phi}$ - from dB magnitude/degree phase noise to magnitude/phase
$\mathbf{v}$	: measurement noise
ϵ	: linear regression equation error
$\boldsymbol{\theta}$	: parameter vector
	$(\bullet)_{GLS}$ - Generalized Least Squares
	$(\bullet)_{GMV}$ - Generalized Minimum Variance
	$(\bullet)_{IV}$ - Instrumental Variable
	$(\bullet)_{LS}$ - Least Squares
	$(\bullet)_{MV}$ - Minimum Variance
	$(\bullet)_{WLS}$ - Weighted Least Squares
ϕ_k	: phase, in rad/sec, of kth frequency response measurement
ϕ_{kdeg}	: phase, in degrees, of kth frequency response measurement
σ	: strength of noise sequence (σ^2 = covariance)
	$(\bullet)_A$ - on real component of frequency response
	$(\bullet)_B$ - on imaginary component of frequency response
	$(\bullet)_{M_{db}}$ - on dB magnitude of frequency response
	$(\bullet)_{\phi_{deg}}$ - on degree phase of frequency response

$(\bullet)_v$ - on plant output

$(\bullet)_w$ - on plant input

AFIT/DS/ENG/96-13

Abstract

Sensor noise is an unavoidable fact of life when it comes to measurements on physical systems, as is the case in feedback control. Therefore, it must be properly addressed during dynamic system identification. In this work, a novel approach is developed toward the treatment of measurement noise in dynamical systems. This approach hinges on proper stochastic modeling, and it can be adapted easily to many different scenarios, where it yields consistently good parameter estimates. The Generalized Minimum Variance algorithm developed and used in this work is based on the theory behind the minimum variance identification process, and the estimate produced is a fixed point of a mapping based on the minimum variance solution. Additionally, the algorithm yields an accurate prediction of the estimation error. This algorithm is applied to many different noise models associated with three basic identification problems. First, continuous-time systems are identified using frequency domain measurements. Next, a discrete-time plant is identified using discrete-time measurements. Finally, the physical parameters of a continuous-time plant are identified using sampled measurements of the continuous-time input and output. Validation of the estimates is performed correctly, and the results are compared with other, more common, identification algorithms. The Generalized Minimum Variance results are generally better than those of the other methods.

COUNTERING THE EFFECTS OF MEASUREMENT NOISE DURING THE IDENTIFICATION OF DYNAMICAL SYSTEMS

I. Introduction

System identification entails an empirical, data driven approach to modeling dynamical systems. The goal of system identification is to determine the parameters of an unknown plant using measurements of the inputs to, and outputs from, the plant. A well-known, but often conveniently ignored, physical reality is that measurement entails uncertainty, and the measurements of the outputs, and possibly the inputs, will be corrupted by an unmeasureable disturbance, *viz.*, *measurement noise.* One should also not lose sight of the purpose of system identification, which can either be used for merely determining the parameter values of interest, or more often, assuming an accurate plant model, for the purpose of designing a control law for controlling the physical plant. Furthermore, there are two realms of system identification. The first case is a one shot, initial identification of an unchanging, but unknown, plant (*e.g.*, for control system design). The other situation is the ongoing system identification of a possibly time varying plant, in which case one refers to adaptive or reconfigurable control, depending on whether the plant's parameters vary slowly or are subject to possibly abrupt change. The latter is of most interest in controls.

In principle, there are two points of view concerning uncertainty and the control of an unknown, or time varying, plant. One is to rely on the benefits of feedback and design a robust feedback control system that is valid for all possible configurations, and the other is to design an adaptive control system that changes its control based on the continuously identified configuration. While (deterministic) robust control can be used to guarantee a level of performance and disturbance rejection, it often leads to high gain controllers which limit performance by saturating the actuators. Moreover, the achieved level of performance for a specific plant realization is oftentimes mediocre. This can be alleviated through adaptive control, where less gain is required, *viz.*, hard actuator saturations will be better accommodated. In addition, superior control performance will be achieved. The need for robust control is not totally eliminated, because the process of system identification is far from perfect. However, online system identification reduces the level of uncertainty, so performance gains can be realized through the use of both robust and adaptive control. Indeed, the adaptive/reconfigurable control approach is most tempting in feedback control, where no new instrumentation and/or actuators beyond those already available for robust control are required. Obviously, the adaptive/reconfigurable control compensator will be nonlinear, as opposed to the linear compensators/controllers that are synthesized in the current linear robust control paradigms. Bearing in mind that modern compensators are, in fact, algorithms, and in view of the ever increasing affordable computer power, the complexity of nonlinear control systems is no longer a restraint, and the adaptive/reconfigurable paradigm is increasingly becoming a viable proposition.

1.1 A Survey of the Literature

System identification is a broad, and hence a very diverse field, for it lies at the heart of the empirical approach to mathematical modeling. For example, a recent issue of *Automatica* was devoted entirely to system identification. However, as the editor states [9], there was a distinct lack of papers on the "application of parameter and system identification". Indeed, only two of the seventeen papers even remotely qualify. This strongly suggests that system identification is still in the process of development. Two major themes present in the current literature are the identification of time varying parameters and the identification of continuous-time models using discrete-time observations. References [21] and [34], respectively, are survey papers that address the current status of these two important lines of research. Additionally, there are many recent papers that propose new methods of dealing with the current difficulties.

In Ref. [5] the authors attempt to identify the physical parameters of a building, modeled by a second-order thermal network, for the purpose of temperature control. Derivatives and integrals of measured variables are used to calculate the recursive least squares estimate of combinations of the physical variables. There is no real validation performed, because the actual parameters are unknown. The data is taken from a real building, and the calculated estimates are believed to be accurate because they are "close" to those estimated in previous work using the same building. As will be pointed out in the sequel, establishing a "correct" validation paradigm in system identification is a crucial element of system identification and is a major theme of this research.

In Ref. [27], continuous-time parameters are once again estimated, but here the observables are passed through a modulating function before being given to the identification algorithm, *viz.*, the input-output pair is filtered. Careful input-output filtering will reduce the effects of measurement noise, and it can result in better estimates. Unfortunately the method requires human intervention for the design and tuning of the filter. Here, the user is required to choose the "modulating frequency index." The estimation algorithm is correctly validated by running a simulation with known parameters and measurement noise, and the estimated parameters are compared to the known parameters.

Another proposed method of identifying continuous-time parameters is to relate the fractional decomposition of the continuous-time system to several z-transform polynomials [29]. Weighted least squares is used to identify the z-transform parameters, and these are converted back to the continuous-time parameter and initial condition estimates. The continuous-time model identification method uses filtering on the input-output data pair to estimate the models. A continuous-time formulation of an ARMAX (dynamic) model is proposed and either recursive least squares or maximum likelihood is used for estimation. The model is validated by comparing the Bode plots and step and impulse responses of the actual and estimated system.

Turning to the time-varying problem, in Ref. [23] weighted least squares is used to examine the nonasymptotic properties of finite memory identification as it relates to time-varying parameters. This work is based on an ARX (static) model (which is not adequate for dynamic system identification), and it discusses both forgetting factors and moving window estimation.

An improvement to the *ad hoc* forgetting factor approach is proposed in Ref. [31]. In this work, an online estimation of the equation error covariance matrix is performed and this estimate is used in the recursive estimation of discrete-time parameters subject to jumps.

The authors of Ref. [35] are aware of the problems associated with weighted recursive least squares for time-varying parameter estimation, and they propose a new algorithm based on a different parameter model. Rather than model the parameter as a constant, they model it as locally changing, and estimate the characteristics of the model. The proposed algorithm contains a pair of user-chosen tuning parameters that allow it to outperform the variable forgetting factor recursive least squares algorithm when applied to a discrete-time ARX model.

Model order determination is also a subject of interest, and the authors of Ref. [24] suggest an innovative method of estimating the system order in conjunction with the parameters. They augment and rearrange the parameter vector and regression matrix before performing an LDL^T factorization of the augmented regression matrix, in which the L matrix contains the parameter estimates for all chosen model orders, and D gives the information on which model order is most correct. Once again, the assumptions made are consistent with an ARX model.

In Ref. [18], the author discusses various methods of parameter estimation, including maximum likelihood estimation, extended Kalman filters, and linear regression as they apply to the identification of correct models for high performance aircraft. Attention is given to determination of proper inputs, correct model order, and model validation.

Measurement of state derivatives are used in Ref. [6] to convert portions of the parameter identification into a "static" process. However low pass filters, with user chosen roll-off frequencies, are used elsewhere in discretization equations for additional estimation. These estimates are then used in an adaptive control law for spacecraft tracking.

1.2 Shortcomings of the Current Identification Paradigm

There are at least three primary methods of identification. Maximum Likelihood Estimation (MLE), which is not considered in this research, is a sound method based on optimization. It does, however, suffer from the process of transcribing a difficult problem into an "equivalent" optimization problem, which requires determination of the *global* minimum. System identification is one area where the global minimum is a requirement, because any local minima, even those close in value to the global minimum, simply produce incorrect parameter estimates. Unfortunately, global optimization is a difficult problem in applied mathematics.

Another classical route to linear system parameter identification entails the estimation of the state of an augmented and nonlinear dynamical system, turning system identification into a nonlinear filtering problem. Hence, it would appear that system identification is in the realm of the Extended Kalman Filter (EKF)[25]. In Extended Kalman Filtering, a linearization is employed. However, when the state estimation error becomes large, this linearization-based approach loses validity and the estimation algorithm fails. One then refers euphemistically to EKF "divergence." In addition, the emphasis in Kalman Filtering is on recursive algorithms, and very often, the complete measurement time history is used. While the recursive approach to estimation is most compatible with using an ever expanding data set, the latter has the deleterious effect of precluding the estimation of time-varying parameters, and in particular, the parameters subject to jumps, as is the case in systems subject to possible failures. Moreover, EKF's must be initialized. When no prior information about the system's states and parameters is available, and the filter is initialized accordingly, it might take a long time for the erroneous information to be "washed out." Furthermore, proper identification requires a data window of minimal physical (temporal) length. Estimates produced using windows shorter than this minimal length, even though high sampling rates can produce many data points, are useless. In conclusion, this identification method suffers from the well known deficiencies of EKF's.

Therefore, in the control community, linear regression-based approaches to the identification of linear system parameters are sought. These are also referred to as "linear prediction" algorithms. Here, the linear structure of the dynamics is directly exploited and the system parameters only (without the states) are estimated. This main line of research in system identification is based on the statistical method of linear regression. This pragmatic approach to system identification is also the subject of this research. The interest in regression-based system identification arises from the relatively easier (compared to MLE) computational burden and its relatively reduced sensitivity to, or possibly no need for, an initial parameter guess. The latter holds the promise of autonomous operation, without the need for human intervention, also known as "tuning".

The main theory behind regression-based identification is least squares. There are many variations on this theme: weighted least squares, generalized least squares, instrumental variables, minimum variance, *etc.* Each suffers from its own difficulties when the estimation problem is not properly formulated. Unfortunately, there is a glaring weakness in the whole of system identification today: the lack of proper treatment of measurement noise. Least Squares (LS), which is widely used in statistics, is a sound method as long as the assumptions upon which it is based are met. Unfortunately, as will be discussed in the sequel, the identification of *dynamic* systems with *measurement* noise negates an important noncorrelation assumption in LS.

The problem is further aggravated by the lack of proper treatment of dynamical system identification with measurement noise in some of the most cited textbooks. When regression based identification is discussed in Refs. [12], [14], and [20], measurement noise is barely mentioned as the discussion moves from least squares theory to dynamical system identification. Unfortunately, the presence of measurement noise in dynamical systems produces a correlation that biases the LS estimate. The LS variations mentioned above were formulated to address the correlation problem, and it could be assumed that measurement noise is just lumped together with other correlated noises. However, this is not a judicious course of action, because measurement noise introduces a specific correlation that *can* be modeled and successfully addressed.

Another misplaced emphasis in the literature is on recursive estimation. The benefits of this methodology are somewhat exaggerated. It is true that the ongoing estimation of a parameter vector could possibly be accomplished faster, and with decreased memory requirements, with a recursive method, but care must be taken in the application of such methods. In recursive estimation, an initial guess is required as to the value of the parameter and the confidence in the guess. If there is no *a priori* knowledge of the parameter, typically a random guess is made, and the confidence in it is set very low. This may seem like a proper thing to do, but poor initial guesses can take a long time to wash out of the recursively calculated estimate. Another fallacy of the recursive methods is that one obtains an estimate after the first measurement, but this estimate is, in fact, meaningless. A certain number of measurements, determined by a minimal *physical* identification interval, is required to obtain a *valid* estimate, and that number does not decrease when using recursive methods. In fact, it could increase because of the poor initial guess. At the same time, the recursive formulation of an identification algorithm oftentimes obscures important aspects of its operation. This aspect of useful estimates will also be addressed in the proposed research.

The identification of time-varying parameters is often based on recursive estimation. The fact that the parameters vary will produce an incorrect estimate as time progresses. This is because the method of recursion, by its nature, remembers all the information. An *ad hoc* method of "losing" the initial information is to introduce a forgetting factor into the recursion. If it is true that the parameter is varying, a better method of estimation would be a moving identification window. The latter method is more amenable to analysis, and the window length can be more directly determined based on information about the physical parameter's drift.

There are two methods of continuous-time system parameter identification [34]. One is to estimate the parameters of a discretized (in time) system using well-known regression techniques, and then convert the discrete-time system back into a continuous-time system using a (bilinear) transformation. A problem arises here, which is addressed in the dissertation, concerning the behavior of the sampled input signal. After all, the continuous physical plant is subject to a continuous input signal, whereas only its sampled values are available to the identification algorithm. The other method of continuous-time system identification entails modeling the continuous parameters directly and then generating certain "measures" from the discretely observed input and output that can be related to the parameters in a regression-based manner. A troublesome aspect of this method is the generation of these "measures." It requires human intervention to select certain parameters involved. Generating measures is equivalent to "properly" filtering the input/output pair. A truly rigorous treatment of the continuous-time case, with continuous-time measurements, resides in the realm of the difficult mathematical theory of nonlinear filtering.

1.3 Research Objectives

One of the main themes of this dissertation is that system identification is not merely an algorithm which can be universally and blindly applied with no prior knowledge. Thus, the main tenant of this work is that system identification is, in fact, a process where the following factors play a major role:

1. The recognition that measurement noise is ubiquitous in physical systems

2. The availability of prior information about the possible system order and parameter values and confidence bounds on the latter

3. Relevant dynamics and the bandwidth of interest

4. The presence of parasitic dynamics, both high and low frequency

5. The way in which these parasitic dynamics interact with the dynamics of interest

6. Excitation issues, including:

   - Physical duration of the identification interval T

   - Sampling rate $1/\Delta T$ or number of measurements $m = T/\Delta T$

   - The number of identified parameters must be small as compared to m (use lower order models if possible)

   - Signal to Noise Ratio (SNR)

   - Frequency content of the input signal

The main thrust of this research is to develop a "correct" and comprehensive system identification paradigm which addresses the items above. To this end, proper stochastic modeling of the measurement noise is of paramount importance. Initially, the theory of minimum variance estimation is applied in an innovative way, based on the correct noise model, and examined in the face of input and output measurement noise. Further subjects of interest are the application of this concept to:

1. Phasor-based system identification of continuous-time systems

2. Static identification of continuous-time systems

3. Time domain identification of discrete-time systems

Although the estimation is performed with discrete, noise-corrupted measurements, the primary goal of this research is to identify the physical parameters of a plant. For that, the continuous-time model is required.

Moreover, in this work, the emphasis is on system identification under the following restrictions:

- Small sample: Proof of asymptotic results (as $m \to \infty$) is quite useless. For one thing, the noise is then washed out. Furthermore, in practice, the small sample identification horizon is typically too short for the asymptotic results to have any bearing on the problem.

- Real time operation: The purpose of the proposed identification is for adaptive control. Therefore, regression-based system identification, which can be adapted to online operation, is used, rather than maximum likelihood estimation.

- Autonomous operation: Identification will, ideally, be performed without human intervention and tuning.

- Measurement noise: It is understood that this is a reality of physical system identification.

The primary consequence of the restrictions above is that the number of identified parameters must be kept small. It is impossible to identify large numbers of parameters accurately with a small sample and significant measurement noise.

Finally, above and beyond the system identification algorithm development work, attention is given to the validation process, especially in the realistic case where one operates on real data.

1.4 Methodology

To achieve these objectives, a novel system identification algorithm is developed and used in this work. Here, the algorithm is coined Generalized Minimum Variance (GMV), because of similarities to the Generalized Least Squares algorithm, and the use of Minimum Variance estimation in the solution. Introduced in this work is the concept that the parameter estimate provided by the GMV algorithm is a fixed point of a nonlinear mapping derived from Minimum Variance estimation. The existence of a fixed point is proven, and the convergence properties of the algorithm are examined for difficult identification problems with low signal-to-noise ratios.

The novel GMV algorithm was first used in Ref. [4] for determining the optimal inputs for the identification of discrete-time plants with output measurement noise only. This noise scenario is examined in this work, and expanded to include input measurement noise as well.

In addition to the discrete-time identification problem, the GMV algorithm is applied to the frequency domain identification of a continuous-time plant, using three different noise scenarios. Also examined in this work is the identification of the parameters of a physical system using samples of the continuous-time inputs and output. In each case, the performance of the GMV algorithm is examined under decreasing signal-to-noise ratios. Careful and well-documented tests are performed, and the results are compared to several, more common, system identification algorithms.

1.5 Organization of the Dissertation

Chapter II focuses on the idea of linear regression (or linear prediction) for system identification. The basic theory of Least Squares is discussed, along with the introduction of correlated equation error. The latter is a result of measurement error incurred during work with dynamical systems, and is revealed by proper modeling of the measurement process. Proper treatment of the equation error correlation is then developed, and the theory behind Minimum Variance identification is applied to the problem. This leads to the GMV algorithm.

Chapter III contains the application of the GMV algorithm to the identification of continuous-time transfer functions using frequency domain measurements. Several different measurement models are considered. These include a basic, but unlikely, model used for comparison purposes, and a realistic measurement model that reflects the capabilities of current frequency analyzers. The GMV identification results are compared to the results of other more common identification methods.

Chapter IV is a short discussion of the effects of unmodeled dynamics on the frequency domain-based identification method. For example, current flight control system design is often performed on low order models, when in fact the true plant contains both unmodeled low and high order dynamics.

Chapter V contains the application of the GMV algorithm to the identification of discrete-time dynamical systems using discrete-time measurements. The assumption here is that the plant is being controlled with a zero-order hold, so the continuous-time parameters can be obtained simply by performing an inverse zero-order hold transformation. The GMV algorithm results are compared at each step to those of alternative, more common, identification schemes. The effects of initial transients, input and output noise, and sampling rate are all investigated here.

Chapter VI is a discussion of the identification of a continuous-time system's physical parameters using sampled measurements of the system's input and output. This work is based on a competition that originated in Italy. The basic dynamical system being identified is based on the thermal characteristics of buildings, but it is basically modeled as a simple second-order electrical circuit.

Finally, chapter VII is a summary and discussion of the results of the research. This is a rich field of work, and this research should only be a beginning.

II. Linear Regression

In the system identification paradigm, there is an observable Z, that is related to an unknown, but desired, parameter vector $\boldsymbol{\theta} = [\theta_1 \ldots \theta_n]^T$. If $\boldsymbol{\theta}$ is constant, and that relation is linear, then a number of observations (m) can be taken

$$Z_k = H_{k1}\theta_1 + H_{k2}\theta_2 + \cdots + H_{kn}\theta_n \quad i = 1, 2, \ldots, m \tag{2.1}$$

where (2.1) is referred to as a *regression function* [14]. These observations can then be arranged in a matrix form linear regression

$$\mathbf{z} = \mathbf{H}\boldsymbol{\theta}, \quad \text{where} \quad \mathbf{z} = \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_m \end{bmatrix} \quad \text{and} \quad \mathbf{H} = \begin{bmatrix} H_{11} & H_{12} & \cdots & H_{1n} \\ H_{21} & H_{22} & \cdots & H_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ H_{m1} & H_{m2} & \cdots & H_{mn} \end{bmatrix}$$

Here $\mathbf{H}$ is referred to as the *regression matrix.* There are many methods to estimate $\boldsymbol{\theta}$ based on the observations, and most are related to Least Squares (LS).

2.1 Least Squares and Weighted Least Squares

The estimation method of least squares was originated by Gauss in the early 19th century, when he used it to predict the orbits of planets. Since then, it has become a commonly used tool in parameter estimation. Its popularity among engineers stems from the fact that it is easier to understand than some other methods, such as maximum likelihood, and that it does not require a knowledge of mathematical statistics [14]. "Deterministic" optimization approaches abound. Also, for certain problems that are formulated properly, the least squares method yields estimates which are consistent, unbiased, and efficient.

If there was no uncertainty in the observations (Z_k and H_{ki} obtainable), only $m = n$ linearly independent observations would be needed to calculate the parameters uniquely. However, "all our measurements and observations are nothing more than approximations to the truth [10]," so there is uncertainty present, and this requires one to make $m > n$ observations. At the very least, there is uncertainty about the Z_k's, *i.e.*, one can obtain an uncertain $z_k = Z_k + v_k$. Therefore,

$$z_k = H_{k1}\theta_1 + H_{k2}\theta_2 + \cdots + H_{kn}\theta_n + v_k$$

which implies that

$$\mathbf{z} = \mathbf{H}\boldsymbol{\theta} + \boldsymbol{\epsilon} \quad \text{where} \quad \boldsymbol{\epsilon} = [v_1 \ldots v_m]^T \quad \text{and} \quad \mathbf{z} = [z_1 \ldots z_m]^T$$

In least squares theory, the objective is to find the estimate $\widehat{\boldsymbol{\theta}}$ which minimizes the square of the equation error vector $\boldsymbol{\epsilon}(\boldsymbol{\theta}) = \mathbf{z} - \mathbf{H}\boldsymbol{\theta}$, *i.e.*, [14]

$$\widehat{\boldsymbol{\theta}}_{LS} = \arg\min_{\boldsymbol{\theta}} \boldsymbol{\epsilon}(\boldsymbol{\theta})^T \boldsymbol{\epsilon}(\boldsymbol{\theta}) = \left(\mathbf{H}^T\mathbf{H}\right)^{-1} \mathbf{H}^T\mathbf{z} \tag{2.2}$$

A basic assumption in the above formulation is that the uncertainty in each of the observations is the same. If it is not, then a weighting matrix $\mathbf{W}$ can be introduced to weight each observation differently, and the Weighted Least Squares (WLS) estimate is then given by

$$\widehat{\boldsymbol{\theta}}_{WLS} = \arg\min_{\boldsymbol{\theta}} \boldsymbol{\epsilon}(\boldsymbol{\theta})^T \mathbf{W} \boldsymbol{\epsilon}(\boldsymbol{\theta}) = \left(\mathbf{H}^T\mathbf{W}\mathbf{H}\right)^{-1} \mathbf{H}^T\mathbf{W}\mathbf{z} \tag{2.3}$$

If some prior knowledge of $\boldsymbol{\epsilon}$ exists, specifically that $\boldsymbol{\epsilon}$ is a stationary random vector with $E[\boldsymbol{\epsilon}] = 0$ and $E[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T] = \mathbf{R}$, where $E[\cdot]$ denotes statistical expectation, then

$$\widehat{\boldsymbol{\theta}}_{MV} = \left(\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\right)^{-1} \mathbf{H}^T\mathbf{R}^{-1}\mathbf{z} \tag{2.4}$$

is the minimum Best Linear Unbiased Estimate (BLUE) [12], or the estimate which has the minimum variance out of all linear unbiased estimates. Therefore, it is referred to as the Minimum Variance (MV) estimate. For the special case when the v_k, $k = 1, \ldots, m$, are identically distributed and independent (i.i.d.) with variance σ^2, the covariance of the estimation error is a scaled identity matrix ($\mathbf{R} = \sigma^2\mathbf{I}$), and the estimate $\widehat{\boldsymbol{\theta}}_{MV} = \widehat{\boldsymbol{\theta}}_{LS}$. Alternatively, if one assumes that v_i are i.i.d. and Gaussian, then $\widehat{\boldsymbol{\theta}}_{MV}$ is the maximum likelihood estimate of $\boldsymbol{\theta}$ [12].

The primary drawback of LS estimation is the requirement that $\mathbf{R} = \sigma^2\mathbf{I}$. If it is not, as in the case when *measurement* noise is present and *dynamical* systems are considered, *i.e.*, the traditional control situation, then $\widehat{\boldsymbol{\theta}}_{LS}$ will be biased. The WLS method eliminates the bias, but $\mathbf{R}$ is often not known in practice. One of the main thrusts in this research is the correct modeling of $\mathbf{R}$, yielding parameter estimates with smaller bias.

2.2 Correlation Caused by Dynamical Systems

In this work, the primary interest lies in the identification of dynamical systems, and in this type of problem, not only are the Z_k's unobtainable, but also the H_{ki}'s. Now, the obtainable quantity is $h_{ki} = H_{ki} + w_{ki}$, and the pertinent equation is

$$z_k = h_{k1}\theta_1 + h_{k2}\theta_2 + \cdots + h_{kn}\theta_n + v_k - w_{k1}\theta_1 - w_{k2}\theta_2 - \cdots - w_{kn}\theta_n \tag{2.5}$$

Here, $\boldsymbol{\epsilon} = [\epsilon_1 \ldots \epsilon_m]^T$ where $\epsilon_k = v_k - \sum_{j=1}^{n} \theta_i w_{ki}$. It is known that, as $m \rightarrow \infty$, the least squares estimate converges as [12]

$$\widehat{\boldsymbol{\theta}}_{LS} \longrightarrow \boldsymbol{\theta} + E[\mathbf{H}\mathbf{H}^T]^{-1}E[\mathbf{H}^T\boldsymbol{\epsilon}] \tag{2.6}$$

If $\mathbf{H}$ and $\boldsymbol{\epsilon}$ are uncorrelated, then $E[\mathbf{H}^T\boldsymbol{\epsilon}] = \mathbf{H}^T E[\boldsymbol{\epsilon}] = \mathbf{H}^T\mathbf{0} = \mathbf{0}$, but for the system given in Eq. (2.5), the kth element of $E[\mathbf{H}^T\boldsymbol{\epsilon}]$ is given by

$$E\left[\sum_{i=1}^{m} h_{ik}\left(v_k - \sum_{j=1}^{n}\theta_j w_{kj}\right)\right]$$

Even if $E[w_{ki}w_{kj}] = \delta_{ij}\sigma^2$, substituting for h_{ik} above yields

$$E\left[\sum_{i=1}^{m}(H_{ik} + w_{ik})\left(v_k - \sum_{j=1}^{n}\theta_j w_{kj}\right)\right] = m\sigma^2\theta_k \neq 0$$

and this correlation causes the LS-based estimate to be biased. Hence, when measurement noise-corrupted dynamical models are used, it is important to recognize the correlation inherent in the linear regression's equation error. It is therefore wise to estimate using a method which incorporates and models the equation error covariance information.

2.3 Dealing with Correlation

There are several methods in the literature today that were developed to overcome the correlation introduced by the dynamical model, thereby reducing the bias in the LS estimate. The two mentioned here are the Generalized Least Squares (GLS) and the Instrumental Variable (IV) methods. However, these methods do not specifically address the *particular* correlation introduced by measurement noise in dynamical systems.

2.3.1 Generalized Minimum Variance. To address the issue of measurement noise properly, the following procedure is offered. The available information can be arranged in a linear regression equation given by

$$\mathbf{z} = \mathbf{H}\boldsymbol{\theta} + \boldsymbol{\epsilon} \qquad \boldsymbol{\epsilon} = \mathbf{T}(\boldsymbol{\theta})\mathbf{v}, \qquad \mathbf{v} = \mathcal{N}(\mathbf{0}, \mathbf{R}_v), \qquad \mathbf{R}_v \geq 0 \tag{2.7}$$

To use insight gained from the minimum variance identification method, it is necessary to obtain the equation error covariance $\mathbf{R}$, where

$$\mathbf{R} = E[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T] = \mathbf{T}\mathbf{R}_v\mathbf{T}^T \tag{2.8}$$

Unfortunately, $\mathbf{R}$ is not known *a priori*, because in addition to the dependence on the given sensor's measurement error σ_v, it is a function of the (as yet unknown) coefficients of the system's transfer function, *i.e.*, $\mathbf{R} = \mathbf{R}(\boldsymbol{\theta})$. Two related, but different, estimates can be derived from this minimum variance based approach. First, one could minimize the associated cost function to obtain the estimate:

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \boldsymbol{\epsilon}(\boldsymbol{\theta})^T\boldsymbol{\epsilon}(\boldsymbol{\theta}) = (\mathbf{z} - \mathbf{H}\boldsymbol{\theta})^T \mathbf{R}^{-1}(\boldsymbol{\theta})\,(\mathbf{z} - \mathbf{H}\boldsymbol{\theta}) \tag{2.9}$$

Because of the dependence of $\mathbf{R}$ on $\boldsymbol{\theta}$, this leads to a complicated numerical search for a global minimum. In global nonlinear searches such as this, the appearance of local minima is virtually guaranteed as the noise levels increase. There is a problem with local minima in system identification problems that tends not to be a problem in other minimization problems. While a local minimum in a general minimization problem probably produces a solution with a lower cost, which may be acceptable as a solution, the local minimum in a system identification problem can be very far from the required global minimum and is probably incorrect. Additionally, in general problems, a local minimum with a cost that is very close to the global minimum cost probably produces an acceptable solution. However, the fact that the cost at a local minimum in a system identification problem is close to the global minimum cost has no bearing on the "closeness" of the estimate to the correct solution. It most likely results in a completely different, and incorrect, estimate. Therefore, only the global minimum is an acceptable solution in system identification.

In an attempt to avoid this complication, Eq. (2.4) is used to obtain the second possible derivation of the estimate. The Generalized Minimum Variance (GMV) estimate is given by the point $\widehat{\boldsymbol{\theta}}_{GMV}$ such that

$$\widehat{\boldsymbol{\theta}}_{GMV} = \left(\mathbf{H}^T\mathbf{R}^{-1}(\widehat{\boldsymbol{\theta}}_{GMV})\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{R}^{-1}(\widehat{\boldsymbol{\theta}}_{GMV})\mathbf{z} \tag{2.10}$$

There are many different ways of searching for fixed points like $\widehat{\boldsymbol{\theta}}_{GMV}$, but the following simple iterative algorithm has been effective in finding the correct fixed point for signal-to-noise ratios approaching 0 dB. This algorithm is motivated by the iteration for fixed points of contraction mappings, for which the existence of a fixed point is guaranteed [1].

Step 1 - Set $i = 0$ and calculate an initial parameter estimate using LS.

$$\widehat{\boldsymbol{\theta}}_i = \widehat{\boldsymbol{\theta}}_0 = \left(\mathbf{H}^T\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{z}$$

Step 2 - Calculate $\mathbf{R}(\widehat{\boldsymbol{\theta}}_i)$ using Eq. (2.8).

Step 3 - Calculate $\widehat{\boldsymbol{\theta}}_{i+1}$ via Eq. (2.4).

$$\widehat{\boldsymbol{\theta}}_{i+1} = \left(\mathbf{H}^T\mathbf{R}^{-1}(\widehat{\boldsymbol{\theta}}_i)\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{R}^{-1}(\widehat{\boldsymbol{\theta}}_i)\mathbf{z}$$

Step 4 - If $\| \widehat{\boldsymbol{\theta}}_{i+1} - \widehat{\boldsymbol{\theta}}_i \|$ is less than some acceptable value, proceed to step 5. Otherwise, increment i and return to step 2.

Step 5 - Set $\widehat{\boldsymbol{\theta}}_{GMV} = \widehat{\boldsymbol{\theta}}_{i+1}$.

Step 6 - The error covariance of the estimate $\widehat{\boldsymbol{\theta}}_{GMV}$ is then given by

$$\left(\mathbf{H}^T\mathbf{R}^{-1}(\widehat{\boldsymbol{\theta}}_{GMV})\mathbf{H}\right)^{-1} \tag{2.11}$$

A fixed point for Eq. (2.10) does exist, at least for the problems of linear dynamical systems, as is shown in Section 3.3. Also, the algorithm has converged within the numerical limits of Matlab for all problems examined thus far. The number of iterations required is quite small for small noise levels, but does increase as the noise level increases.

Problems do arise as the noise levels increase, as is common in system identification problems at low signal-to-noise ratios. In this work, it is common for one other fixed point to appear at higher noise levels, and possible methods of dealing with this are discussed in Sections 3.6.3 and 5.4. However, the other methods of identification used for comparison in this work tend to suffer at these noise levels as well.

2.3.2 Other Methods of Correlation Compensation. There are many methods of compensating for correlated measurement noise, as discussed in [30], but the two used for comparison in this work are the Instrumental Variable (IV) method and the Generalized Least Squares (GLS) method. These two methods, or variations thereof, are common in the literature. One problem with each of them is that the derivation tends to be specific to the given problem, so the particular method must be discussed in context, as is done in the sequel.

Briefly put, however, the IV method uses the fact that the Z_k's and H_{ki}'s are actually functions of some fixed, but unknown $\boldsymbol{\theta}$. During the IV identification method, the current estimate is used to produce a set of estimated $\widehat{H}_{ki}$'s which are then used in an iterative scheme where

$$\widehat{\boldsymbol{\theta}}_{i+1} = \left(\widehat{\mathbf{H}(\widehat{\boldsymbol{\theta}}_i)}^T \mathbf{H}\right)^{-1} \widehat{\mathbf{H}(\widehat{\boldsymbol{\theta}}_i)}^T \mathbf{z}$$

Similarly, the GLS method exploits the form of $\boldsymbol{\epsilon}$ in Eq. (2.7). The contents of the $\mathbf{z}$ and $\mathbf{H}$ matrices are filtered through some form of $\mathbf{T}^{-1}$, and the LS estimate is calculated using these filtered results. The filtering tends to "whiten" the problem, reducing the correlation and causing the LS formulation to be more correct. This is also an iterative procedure, as $\mathbf{T}$ is a function of $\boldsymbol{\theta}$, so the contents of $\mathbf{z}$ and $\mathbf{H}$ must be re-filtered at each step.

2.4 Conclusions

The correlation introduced by dynamical systems and measurement noise effectively eliminates the LS estimate as a valid result. Some method of dealing with this correlation must be found in order to obtain an accurate estimate. There are many current methods of dealing with the correlation, but none seem to specifically address the particular correlation introduced by measurement noise.

The stochastic modeling and GMV identification concept described in this chapter are not limited to time-domain dynamical systems. They can also be applied to any problem where certain measurements are correlated with one another. An example of this is the frequency domain identification of a continuous-time plant, which is discussed in the next chapter.

III. Phasor-Based Identification of a Continuous-Time Dynamical System

In this chapter, a phasor approach to system identification is discussed, *viz.*, the experimental data consists of a finite number of sensor noise corrupted point frequency response measurements of the unknown plant.

The proposed identification paradigm is in line with currently available instrumentation, *e.g.*, frequency analyzers [33]. Gaussian measurement noise statistics are assumed. The emphasis here is on a stochastic analysis of the identification problem in the frequency domain. Parameter estimation algorithms are developed and validated in simulations which include measurement noise. Careful modeling of the stochastic estimation problem renders an efficient system identification algorithm.

Moreover, the proposed system identification algorithm yields an identified model and an estimate of the model uncertainty. The error is expressed in terms of the uncertainty of the coefficients of the plant's transfer function, the latter being easily transformable into an expression of uncertainty about the physical parameters of the plant. In other words, the proposed system identification algorithm directly yields an identified model and its structured uncertainty, as opposed to plant uncertainty gauged with the H_∞ norm. Furthermore, the H_∞ estimation error can be calculated using optimization methods, while Kharitonov [17] type results can be invoked to ascertain robust stability. Obviously, the measurement error is somewhat indicative of the H_∞ estimation error.

The phasor approach is developed in Section 3.1. A stochastic analysis of the system identification problem with measurement noise is performed in Section 3.2. Section 3.3 contains a proof that at least one fixed point exists for this formulation. Section 3.4 explains how to calculate a needed covariance matrix for different assumptions about the measurement model. A brief discussion of the other algorithms being compared is contained in Section 3.5. Sections 3.6 contains the results of applying the GMV algorithm to a second-order system, as well as the comparisons to the other algorithms. The system used is representative of the short period dynamics of an aircraft. The performance of the different system identification algorithms is briefly discussed in Section 3.7, and concluding remarks are made in Section 3.8.

3.1 The Frequency Domain Based Phasor Approach

Consider the stable frequency domain transfer function, where the order of the numerator and denominator are known:

$$T(s) = \frac{y(s)}{u(s)} = \frac{b_1 s^{n-1} + b_2 s^{n-2} + \cdots + b_{n-1}s + b_n}{s^n - a_1 s^{n-1} - a_2 s^{n-2} - \cdots - a_{n-1}s - a_n} \qquad (3.1)$$

Here, the parameters $a_1 \ldots a_n$ and $b_1 \ldots b_n$ are unknown, the $a_1 \ldots a_n$ coefficients are negative for a stable system, and the $b_1 \ldots b_{q-1}$ coefficients equal zero when the order of the numerator is q less than the order of the denominator. If these parameters are equal to zero, it should be taken into account when the problem is developed. If one attempts to blindly identify all the parameters, assuming that the zero value parameters will just identify to zero, the results will suffer.

By applying an input to this system of the form

$$u_k(t) = cos(\omega_k t)$$

and waiting for steady-state to be achieved, the steady-state output is

$$y_k(t) = (A_k + jB_k)cos(\omega_k t)$$

The following is a fundamental result of linear system theory.

$$A_k + jB_k = \frac{b_1(j\omega_k)^{n-1} + \cdots + b_i(j\omega_k)^{n-i} + \cdots + b_{n-1}(j\omega_k) + b_n}{(j\omega_k)^n - a_1(j\omega_k)^{n-1} - \cdots - a_i(j\omega_k)^{n-i} - \cdots - a_{n-1}(j\omega_k) - a_n} \qquad (3.2)$$

Hence, a linear equation in the $2n$ a_i and b_i coefficients is obtained

$$(A_k + jB_k)(j\omega_k)^n = \sum_{i=1}^{n} a_i(A_k + jB_k)(j\omega_k)^{n-i} + \sum_{i=1}^{n} b_i(j\omega_k)^{n-i}$$

which yields

$$(A_k + jB_k) = \sum_{i=1}^{n}(A_k + jB_k)(j\omega_k)^{-i}a_i + \sum_{i=1}^{n}(j\omega_k)^{-i}b_i$$

Through algebraic manipulations, one obtains the linear system:

$$\left[\begin{array}{c} \overbrace{\begin{matrix} \frac{A_k}{\omega_k} & \frac{B_k}{\omega_k^2} & \frac{-A_k}{\omega_k^3} & \frac{-B_k}{\omega_k^4} & \frac{A_k}{\omega_k^5} & \cdots \\ \frac{B_k}{\omega_k} & \frac{-A_k}{\omega_k^2} & \frac{-B_k}{\omega_k^3} & \frac{A_k}{\omega_k^4} & \frac{B_k}{\omega_k^5} & \cdots \end{matrix}}^{n} \overbrace{\begin{matrix} \frac{1}{\omega_k} & 0 & \frac{-1}{\omega_k^3} & 0 & \frac{1}{\omega_k^5} & \cdots \\ 0 & \frac{-1}{\omega_k^2} & 0 & \frac{1}{\omega_k^4} & 0 & \cdots \end{matrix}}^{n} \end{array}\right] \begin{bmatrix} a_1 \\ \vdots \\ a_n \\ b_1 \\ \vdots \\ b_n \end{bmatrix} = \begin{bmatrix} -B_k \\ A_k \end{bmatrix} \tag{3.3}$$

As can be seen, each sinusoidal test input produces two equations in the $2n$ unknowns. Therefore, n distinct sinusoids are needed to produce the $2n$ equations required for the determination of the parameter vector $\boldsymbol{\theta} \equiv [a_1 \ \ldots \ a_n \ b_1 \ \ldots \ b_n]^T \in \mathcal{R}^{2n}$.

3.2 Measurement Noise

In reality, no physical measurements can be made without some sort of measurement noise present, so the true A_k and B_k are not available. Rather, there is the noise corrupted A_{k_m} and B_{k_m}, where the measured quantities are

$$A_{k_m} = A_k + v_{A_k}, \qquad v_A = \mathcal{N}(0, \sigma_A^2)$$

$$B_{k_m} = B_k + v_{B_k}, \qquad v_B = \mathcal{N}(0, \sigma_B^2)$$

Here, the random variables v_{A_k} and v_{B_k} are the measurement noises in the kth experiment, $k = 1, 2, \ldots, m$. Therefore in reality, Eq. (3.3) is

$$\left[\begin{array}{c} \overbrace{\begin{matrix} \frac{(A_{k_m} - v_{A_k})}{\omega_k} & \frac{(B_{k_m} - v_{B_k})}{\omega_k^2} & \frac{-(A_{k_m} - v_{A_k})}{\omega_k^3} & \cdots \\ \frac{(B_{k_m} - v_{B_k})}{\omega_k} & \frac{-(A_{k_m} - v_{A_k})}{\omega_k^2} & \frac{-(B_{k_m} - v_{B_k})}{\omega_k^3} & \cdots \end{matrix}}^{n} \overbrace{\begin{matrix} \frac{1}{\omega_k} & 0 & \frac{-1}{\omega_k^3} & \cdots \\ 0 & \frac{-1}{\omega_k^2} & 0 & \cdots \end{matrix}}^{n} \end{array}\right] \boldsymbol{\theta} = \begin{bmatrix} -(B_{k_m} - v_{B_k}) \\ (A_{k_m} - v_{A_k}) \end{bmatrix} \tag{3.4}$$

This equation can now be written as a regression function with

$$\mathbf{z}_k = \begin{bmatrix} -B_{k_m} \\ A_{k_m} \end{bmatrix}$$

$$\mathbf{H}_k = \left[\overbrace{\begin{matrix} \frac{A_{km}}{\omega_k} & \frac{B_{km}}{\omega_k^2} & \frac{-A_{km}}{\omega_k^3} & \frac{-B_{km}}{\omega_k^4} & \frac{A_{km}}{\omega_k^5} & \cdots \\ \frac{B_{km}}{\omega_k} & \frac{-A_{km}}{\omega_k^2} & \frac{-B_{km}}{\omega_k^3} & \frac{A_{km}}{\omega_k^4} & \frac{B_{km}}{\omega_k^5} & \cdots \end{matrix}}^{n} \overbrace{\begin{matrix} \frac{1}{\omega_k} & 0 & \frac{-1}{\omega_k^3} & 0 & \frac{1}{\omega_k^5} & \cdots \\ 0 & \frac{-1}{\omega_k^2} & 0 & \frac{1}{\omega_k^4} & 0 & \cdots \end{matrix}}^{n} \right]$$

$$\boldsymbol{\theta} = \begin{bmatrix} a_1 & a_2 & \cdots & a_n & b_1 & b_2 & \cdots & b_n \end{bmatrix}^T$$

$$\boldsymbol{\epsilon}_k = \left[\begin{matrix} \overbrace{-\frac{v_{A_k} a_1}{\omega_k} - \frac{v_{B_k} a_2}{\omega_k^2} + \frac{v_{A_k} a_3}{\omega_k^3} + \frac{v_{B_k} a_4}{\omega_k^4} - \frac{v_{A_k} a_5}{\omega_k^5} - \cdots}^{n} & -v_{B_k} \\ -\frac{v_{B_k} a_1}{\omega_k} + \frac{v_{A_k} a_2}{\omega_k^2} + \frac{v_{B_k} a_3}{\omega_k^3} - \frac{v_{A_k} a_4}{\omega_k^4} - \frac{v_{B_k} a_5}{\omega_k^5} + \cdots & +v_{A_k} \end{matrix} \right] \tag{3.5}$$

$$\boldsymbol{\epsilon}_k = \underbrace{\begin{bmatrix} -\frac{a_1}{\omega_k} + \frac{a_3}{\omega_k^3} - \frac{a_5}{\omega_k^5} \cdots & -1 - \frac{a_2}{\omega_k^2} + \frac{a_4}{\omega_k^4} \cdots \\ 1 + \frac{a_2}{\omega_k^2} - \frac{a_4}{\omega_k^4} \cdots & -\frac{a_1}{\omega_k} + \frac{a_3}{\omega_k^3} - \frac{a_5}{\omega_k^5} \cdots \end{bmatrix}}_{\mathbf{T}_k} \underbrace{\begin{bmatrix} v_{A_k} \\ v_{B_k} \end{bmatrix}}_{\mathbf{v}_k} = \mathbf{T}_k \mathbf{v}_k \tag{3.6}$$

Additionally, more than the minimal n sinusoidal test signals will be needed to overcome the measurement noise effects and obtain an accurate estimate. Hence, m ($\geq n$) measurements are taken, and the measurement information is arrayed in a linear regression as follows

$$\begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_m \end{bmatrix}_{2m\times 1} = \begin{bmatrix} \mathbf{H}_1 \\ \mathbf{H}_2 \\ \vdots \\ \mathbf{H}_m \end{bmatrix}_{2m\times 2n} \boldsymbol{\theta} + \begin{bmatrix} \mathbf{T}_1\mathbf{v}_1 \\ \mathbf{T}_2\mathbf{v}_2 \\ \vdots \\ \mathbf{T}_m\mathbf{v}_m \end{bmatrix}_{2m\times 1}$$

This is now in a linear regression form as in Eq. (2.7), where the Gaussian random vector $\boldsymbol{\epsilon}$ is the equation error resulting from the measurement noise. The critical measurement noise covariance matrix $\mathbf{R}$ is analytically determined by calculating the expectation $E\left\{\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T\right\}$. If the measurement noise is uncorrelated from experiment to experiment $\left(E\left\{\mathbf{v}_k\mathbf{v}_j^T\right\} = 0,\ k \neq j\right)$, then $\mathbf{R}$ is reduced to a block diagonal matrix of the form

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_1 & 0 & \cdots & 0 \\ 0 & \mathbf{R}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{R}_m \end{bmatrix}_{2m\times 2m}$$

where the 2×2 $\mathbf{R}_k$ submatrix ($k = 1, 2, \ldots, m$) is of the form

$$\begin{aligned}\mathbf{R}_k &= E\left\{\boldsymbol{\epsilon}_k \boldsymbol{\epsilon}_k^T\right\} = E\left\{\mathbf{T}_k \mathbf{v}_k \mathbf{v}_k^T \mathbf{T}_k^T\right\} \\ &= \mathbf{T}_k \mathbf{R}_{AB} \mathbf{T}_k^T \quad \text{where} \quad \mathbf{R}_{AB} = \begin{bmatrix} \sigma_A^2 & \sigma_{AB} \\ \sigma_{AB} & \sigma_B^2 \end{bmatrix}\end{aligned}$$

Note that

$$\mathbf{R}^{-1} = \begin{bmatrix} \mathbf{R}_1^{-1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \mathbf{R}_m^{-1} \end{bmatrix}, \qquad \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} = \sum_{k=1}^{m} \mathbf{H}_k^T \mathbf{R}_k^{-1} \mathbf{H}_k, \qquad \mathbf{R}^{-1} \mathbf{z} = \sum_{k=1}^{m} \mathbf{R}_k^{-1} \mathbf{z}_k$$

so

$$\begin{aligned}\widehat{\boldsymbol{\theta}}_{GMV} &= \left(\sum_{k=1}^{m} \mathbf{H}_k^T \mathbf{R}_k^{-1}(\widehat{\boldsymbol{\theta}}_{GMV}) \mathbf{H}_k \right)^{-1} \sum_{k=1}^{m} \mathbf{H}_k^T \mathbf{R}_k^{-1}(\widehat{\boldsymbol{\theta}}_{GMV}) \mathbf{z}_k \\ \mathbf{P}_{GMV} &= \left(\sum_{k=1}^{m} \mathbf{H}_k^T \mathbf{R}_k^{-1}(\widehat{\boldsymbol{\theta}}_{GMV}) \mathbf{H}_k \right)^{-1}\end{aligned} \tag{3.7}$$

3.3 Existence of Fixed Point

The primary problem in solving for the GMV estimate is that $\mathbf{R}$ is a nonlinear function of $\boldsymbol{\theta}$. Therefore, define $f : \mathcal{R}^{2n} \to \mathcal{R}^{2n}$,

$$f(\boldsymbol{\theta}) = \left[\mathbf{H}^T \mathbf{R}^{-1}(\boldsymbol{\theta}) \mathbf{H}\right]^{-1} \mathbf{H}^T \mathbf{R}^{-1}(\boldsymbol{\theta}) \mathbf{z} \tag{3.8}$$

and use an iteration to search for a fixed point in $\mathcal{R}^{2n}$ where $\boldsymbol{\theta} = f(\boldsymbol{\theta})$. In searching for a fixed point, it would be nice to know that a fixed point actually exists. If it could be determined that $f(\boldsymbol{\theta})$ is a continuous, bounded function, then it would be easy to show that at least one fixed point exists.

3.3.1 Singularities and Continuity. The first step in showing that $f(\boldsymbol{\theta})$ is bounded is to show that it is a continuous function, or that it has only removable singularities. To do this, first return to the equation error noise vector in Eq. (3.6). Because sums, products, and inverses of rational functions are rational functions, and $\mathbf{T}_k$ is a matrix of rational functions in the parameters, then $f(\boldsymbol{\theta})$ is a rational function. Therefore, $f(\boldsymbol{\theta})$ is continuous everywhere it is defined. However, $f(\boldsymbol{\theta})$ is not defined everywhere, but has singularities where $\mathbf{R}$ is not invertible. Since $\mathbf{R}_{AB}$ is, by definition, positive definite, the singularities occur when the determinant of any particular $\mathbf{T}_k$ matrix evaluates to zero. Since the transformation matrix $\mathbf{T}_k$ is affine in the denominator parameters, it can be written as

$$\begin{aligned}\mathbf{T}_k &= \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} + \begin{bmatrix} -\frac{1}{\omega_k} & 0 \\ 0 & -\frac{1}{\omega_k} \end{bmatrix} a_1 + \begin{bmatrix} 0 & -\frac{1}{\omega_k^2} \\ \frac{1}{\omega_k^2} & 0 \end{bmatrix} a_2 + \begin{bmatrix} \frac{1}{\omega_k^3} & 0 \\ 0 & \frac{1}{\omega_k^3} \end{bmatrix} a_3 + \cdots \\ &= \mathbf{A}_0 + \sum_{j=1}^{n} \mathbf{A}_{jk} a_j \end{aligned} \tag{3.9}$$

From Eq. (3.6), it can be seen that $\mathbf{T}$ always has the form

$$\mathbf{T} = \begin{bmatrix} a & b \\ -b & a \end{bmatrix}$$

This implies that $\det(\mathbf{T}) = a^2 + b^2$, and the only way that this can be zero is if a and b are zero, *i.e.*, $\mathbf{T} = \mathbf{0}$. For the case $n = 1$, it is impossible for $\mathbf{T}_k$ to equal $\mathbf{0}$. For $n = 2$, the only way $\mathbf{T}_k$ can equal $\mathbf{0}$ is if $a_1 = 0$ and $a_2 = -\omega_k^2$. This leads to m distinct singular points of $f(\boldsymbol{\theta})$; one for each measurement frequency. For $n = 3$, if $a_2 = -\omega_k^2$ and $a_3 = \omega_k^2 a_1$, then $\mathbf{T}_k = \mathbf{0}$. This implies that there are m distinct lines of singularities in $\boldsymbol{\theta}$ space, or m distinct singular manifolds of dimension 1. For general n, the dimension of the singular manifolds is $n - 2$.

To get an idea of the nature of these singularities, an $n = 2$ case is considered. Three measurements ($\omega = 1, 2, 10$) are taken of a second-order plant, and $f(\boldsymbol{\theta})$ is plotted along the path $a_1 = 0$. The results in Fig. 3.1 appear to show that the singularities are removable. Indeed, this seems to be the case for higher dimension problems as well. If $f(\boldsymbol{\theta})$ is evaluated at $\tilde{\boldsymbol{\theta}} + \boldsymbol{\epsilon}$, where $\tilde{\boldsymbol{\theta}}$ is a singular point and $\boldsymbol{\epsilon}$ is some small vector, the results are close for any $\boldsymbol{\epsilon}$, implying that the singularity is removable as well. For the singularity to be removable, the following limit must exist.

$$\lim_{\boldsymbol{\theta} \to \tilde{\boldsymbol{\theta}}} f(\boldsymbol{\theta})$$

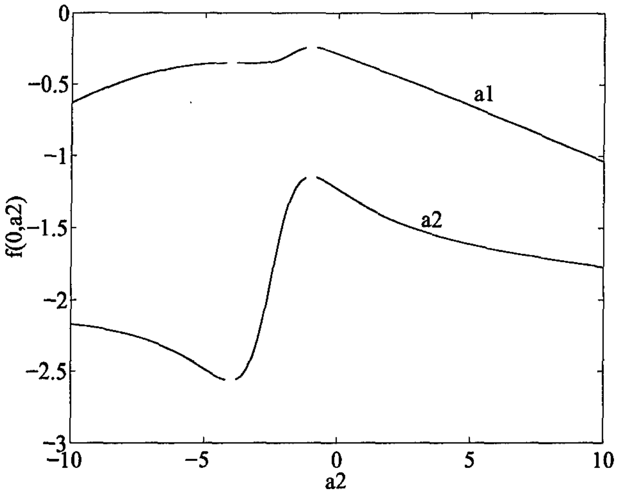


Figure 3.1. $f(\boldsymbol{\theta})$ for $a_1 = 0$

Insight can now be gained by looking at the physical nature of the problem. The singularities occur when $\mathbf{T}_k = \mathbf{0}$, which implies that $\boldsymbol{\epsilon} = \mathbf{0}$ in Eq. (3.6), *i.e.*, there is no uncertainty in that measurement. For $n = 2$, this would mean $\mathbf{z}_{\tilde{k}} = \mathbf{H}_{\tilde{k}}\boldsymbol{\theta}$, and a solution can be found by simple least squares, *i.e.*,

$$\lim_{\boldsymbol{\theta}\to\tilde{\boldsymbol{\theta}}} f(\boldsymbol{\theta}) = \left[\mathbf{H}_{\tilde{k}}^T\mathbf{H}_{\tilde{k}}\right]^{-1}\mathbf{H}_{\tilde{k}}^T\mathbf{z}_{\tilde{k}} \tag{3.10}$$

Unfortunately, this formulation is only valid for $n \leq 2$, because for $n > 2$, $\mathbf{H}_{\tilde{k}}^T\mathbf{R}_{\tilde{k}}^{-1}\mathbf{H}_{\tilde{k}}$ only has rank 2 and is therefore not invertible.

However, one can expand on the notion of a noiseless measurement, and use the Kalman filter update equations to add an additional measurement to a previous estimate. For $n = 2$ and $n = 3$, it is readily apparent that only one $\mathbf{T}_k$ can be equal to $\mathbf{0}$ for a given experiment. For $n = 2$, the singularities are points corresponding to $a_2 = -\omega_{\tilde{k}}^2$, and since two measurements cannot be taken at the same frequency, the points cannot be repeated. For $n = 3$, the individual singularity lines lie in the plane $a_2 = -\omega_{\tilde{k}}^2$, and again, since no two measurements can be taken at the same frequency, the singularity lines cannot intersect. It is suspected that this is the case for higher n as well.

This implies that there is only one "noiseless measurement" at any given point. In other words, only one term in each of the summations is undefined at any given singular point:

$$\mathbf{H}_{\tilde{k}}^T \mathbf{R}_{\tilde{k}}^{-1} \mathbf{H}_{\tilde{k}} \quad \text{and} \quad \mathbf{H}_{\tilde{k}}^T \mathbf{R}_{\tilde{k}}^{-1} \mathbf{z}_{\tilde{k}}$$

These terms are not defined because $\mathbf{T}_{\tilde{k}} = \mathbf{0}$, causing $\mathbf{R}_{\tilde{k}} = \mathbf{0}$. Taking the Kalman filtering approach, let the pre-update covariance and estimate be the covariance and estimate calculated by using all the terms except the $\tilde{k}$th one.

$$\bar{\mathbf{P}} = \left(\sum_{k \neq \tilde{k}} \mathbf{H}_k^T \mathbf{R}_k^{-1} \mathbf{H}_k \right)^{-1} \quad \text{and} \quad \bar{\boldsymbol{\theta}} = \bar{\mathbf{P}} \sum_{k \neq \tilde{k}} \mathbf{H}_k^T \mathbf{R}_k^{-1} \mathbf{z}_k$$

Then following the formulation in Ref. [14], it can be shown that in any neighborhood around the singularity,

$$\begin{aligned}
f(\boldsymbol{\theta}) &= \left[\sum_{k=1}^{m} \mathbf{H}_k^T \mathbf{R}_k^{-1} \mathbf{H}_k \right]^{-1} \left[\sum_{k=1}^{m} \mathbf{H}_k^T \mathbf{R}_k^{-1} \mathbf{z}_k \right] \\
&= \left[\bar{\mathbf{P}}^{-1} + \mathbf{H}_{\tilde{k}}^T \mathbf{R}_{\tilde{k}}^{-1} \mathbf{H}_{\tilde{k}} \right]^{-1} \left[\bar{\mathbf{P}}^{-1} \bar{\boldsymbol{\theta}} + \mathbf{H}_{\tilde{k}}^T \mathbf{R}_{\tilde{k}}^{-1} \mathbf{z}_{\tilde{k}} \right] \\
&= \left[\bar{\mathbf{P}} - \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \left(\mathbf{R}_{\tilde{k}} + \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \right)^{-1} \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \right] \left[\bar{\mathbf{P}}^{-1} \bar{\boldsymbol{\theta}} + \mathbf{H}_{\tilde{k}}^T \mathbf{R}_{\tilde{k}}^{-1} \mathbf{z}_{\tilde{k}} \right] \\
&= \bar{\mathbf{P}} \bar{\mathbf{P}}^{-1} \bar{\boldsymbol{\theta}} - \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \left(\mathbf{R}_{\tilde{k}} + \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \right)^{-1} \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \bar{\mathbf{P}}^{-1} \bar{\boldsymbol{\theta}} \\
&\quad + \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \mathbf{R}_{\tilde{k}}^{-1} \mathbf{z}_{\tilde{k}} - \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \left(\mathbf{R}_{\tilde{k}} + \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \right)^{-1} \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \mathbf{R}_{\tilde{k}}^{-1} \mathbf{z}_{\tilde{k}} \\
&= \mathbf{I} \bar{\boldsymbol{\theta}} - \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \left(\mathbf{R}_{\tilde{k}} + \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \right)^{-1} \mathbf{H}_{\tilde{k}} \mathbf{I} \bar{\boldsymbol{\theta}} \\
&\quad + \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \left[\mathbf{R}_{\tilde{k}}^{-1} - \left(\mathbf{R}_{\tilde{k}} + \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \right)^{-1} \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \mathbf{R}_{\tilde{k}}^{-1} \right] \mathbf{z}_{\tilde{k}} \\
&= \bar{\boldsymbol{\theta}} - \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \left(\mathbf{R}_{\tilde{k}} + \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \right)^{-1} \mathbf{H}_{\tilde{k}} \bar{\boldsymbol{\theta}} \\
&\quad + \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \left(\mathbf{R}_{\tilde{k}} + \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \right)^{-1} \left[\left(\mathbf{R}_{\tilde{k}} + \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \right) \mathbf{R}_{\tilde{k}}^{-1} - \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \mathbf{R}_{\tilde{k}}^{-1} \right] \mathbf{z}_{\tilde{k}} \\
&= \bar{\boldsymbol{\theta}} - \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \left(\mathbf{R}_{\tilde{k}} + \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \right)^{-1} \mathbf{H}_{\tilde{k}} \bar{\boldsymbol{\theta}} \\
&\quad + \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \left(\mathbf{R}_{\tilde{k}} + \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \right)^{-1} \left[\mathbf{R}_{\tilde{k}} \mathbf{R}_{\tilde{k}}^{-1} + \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \mathbf{R}_{\tilde{k}}^{-1} - \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \mathbf{R}_{\tilde{k}}^{-1} \right] \mathbf{z}_{\tilde{k}} \\
&= \bar{\boldsymbol{\theta}} - \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \left(\mathbf{R}_{\tilde{k}} + \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \right)^{-1} \mathbf{H}_{\tilde{k}} \bar{\boldsymbol{\theta}} + \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \left(\mathbf{R}_{\tilde{k}} + \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \right)^{-1} \mathbf{I} \mathbf{z}_{\tilde{k}} \\
&= \bar{\boldsymbol{\theta}} + \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \left(\mathbf{R}_{\tilde{k}} + \mathbf{H}_{\tilde{k}} \bar{\mathbf{P}} \mathbf{H}_{\tilde{k}}^T \right)^{-1} \left(\mathbf{z}_{\tilde{k}} - \mathbf{H}_{\tilde{k}} \bar{\boldsymbol{\theta}} \right)
\end{aligned}$$

Therefore, define $g(\boldsymbol{\theta})$ to be

$$g(\boldsymbol{\theta}) = \bar{\boldsymbol{\theta}} + \bar{\mathbf{P}}\mathbf{H}_{\tilde{k}}^T \left(\mathbf{H}_{\tilde{k}}\bar{\mathbf{P}}\mathbf{H}_{\tilde{k}}^T + \mathbf{R}_{\tilde{k}}\right)^{-1} (\mathbf{z}_{\tilde{k}} - \mathbf{H}_{\tilde{k}}\bar{\boldsymbol{\theta}})$$

As can be seen, $g(\boldsymbol{\theta})$ does not have a singularity when $\mathbf{R}_{\tilde{k}} = 0$, and for any given $\tilde{k}$ singularity, $f(\boldsymbol{\theta}) = g(\boldsymbol{\theta})$ in a deleted neighborhood around that singularity. But $g(\boldsymbol{\theta})$ is defined and continuous on the entire neighborhood, so extending $f(\boldsymbol{\theta})$ by filling in its singularities with $g(\boldsymbol{\theta})$ results in a continuous function that is defined and continuous on the whole of $\mathcal{R}^{2n}$.

3.3.2 Boundedness. It is now necessary to show that $f(\boldsymbol{\theta})$ is a bounded function. To show boundedness for a continuous function, one can show that the limit as $\boldsymbol{\theta}$ grows unboundedly exists and is bounded. First, define

$$\mathcal{H} = \left\{\mathbf{h} \in \mathcal{R}^{2n} \ : \| \mathbf{h} \| = 1\right\}$$

$$\boldsymbol{\theta} = \alpha\mathbf{h} \quad \text{where} \quad \mathbf{h} \in \mathcal{H} \text{ and } \alpha \in \mathcal{R} \tag{3.11}$$

Now for any $\mathbf{h}$, look at the limit of $f(\alpha\mathbf{h})$ as α approaches infinity. Provided that the limits exist,

$$\begin{aligned}\lim_{\alpha\to\infty}\left(\frac{\alpha^2}{\alpha^2}\right) f(\boldsymbol{\theta}) &= \left[\lim_{\alpha\to\infty} \alpha^2 \sum_{k=1}^{m} \mathbf{H}_k^T \mathbf{R}_k^{-1}\mathbf{H}_k\right]^{-1} \left[\lim_{\alpha\to\infty} \alpha^2 \sum_{k=1}^{m} \mathbf{H}_k^T \mathbf{R}_k^{-1}\mathbf{z}_k\right] \\ &= \left[\sum_{k=1}^{m} \mathbf{H}_k^T \left(\lim_{\alpha\to\infty}\frac{\mathbf{R}_k}{\alpha^2}\right)^{-1} \mathbf{H}_k\right]^{-1} \left[\sum_{k=1}^{m} \mathbf{H}_k^T \left(\lim_{\alpha\to\infty}\frac{\mathbf{R}_k}{\alpha^2}\right)^{-1} \mathbf{z}_k\right]\end{aligned}$$

This reduces the problem to finding the limit of $\mathbf{R}_k/\alpha^2$ as α approaches infinity, which is a significantly simpler problem, since

$$\lim_{\alpha\to\infty}\frac{\mathbf{R}_k}{\alpha^2} = \left(\lim_{\alpha\to\infty}\frac{\mathbf{T}_k}{\alpha}\right)\mathbf{R}_{AB}\left(\lim_{\alpha\to\infty}\frac{\mathbf{T}_k}{\alpha}\right)^T$$

But here,

$$\mathbf{T}_k = \mathbf{A}_0 + \alpha\sum_{j=1}^{n}\mathbf{A}_{jk}h_j$$

so

$$\lim_{\alpha\to\infty}\frac{\mathbf{T}_k}{\alpha} = \lim_{\alpha\to\infty}\left(\frac{\mathbf{A}_0}{\alpha} + \sum_{j=1}^{n}\mathbf{A}_{jk}h_j\right) = \sum_{j=1}^{n}\mathbf{A}_{jk}h_j$$

which leads to

$$\lim_{\alpha\to\infty}\frac{\mathbf{R}_k}{\alpha^2}=\left(\sum_{j=1}^{n}\mathbf{A}_{jk}h_j\right)\mathbf{R}_{AB}\left(\sum_{j=1}^{n}\mathbf{A}_{jk}h_j\right)^T\equiv\mathbf{R}_{k\infty}$$

and

$$\lim_{\alpha\to\infty}f(\boldsymbol{\theta})=\left[\sum_{k=1}^{m}\mathbf{H}_k^T\mathbf{R}_{k\infty}^{-1}\mathbf{H}_k\right]^{-1}\sum_{k=1}^{m}\mathbf{H}_k^T\mathbf{R}_{k\infty}^{-1}\mathbf{z}_k\equiv f_\infty(\mathbf{h})$$

Since for any given $\mathbf{h}$, $f_\infty(\mathbf{h})$ is simply the solution to a weighted least squares problem, it is therefore a finite vector, and

$$\| f_\infty(\mathbf{h}) \|_\infty < \infty$$

Now let

$$m(\alpha)=\max_{\mathbf{h}\in\mathcal{H}} f(\alpha\mathbf{h})$$

From Ref. [8], it is known that $m(\alpha)$ is a continuous function of α. Using the vector infinity norm,

$$\begin{aligned}\| m(\alpha_1)-m(\alpha_2) \| &= \| \max_{\mathbf{h}} f(\alpha_1\mathbf{h})-\max_{\mathbf{h}} f(\alpha_2\mathbf{h}) \| \\ &\leq \max_{\mathbf{h}} \| f(\alpha_1\mathbf{h})-f(\alpha_2\mathbf{h}) \| \\ &< \epsilon \quad \text{for} \quad \| \alpha_1-\alpha_2 \| < \delta\end{aligned}$$

Also,

$$\lim_{\alpha\to\infty} m(\alpha)=\max_{\mathbf{h}\in\mathcal{H}} f_\infty(\mathbf{h})<\infty$$

Since $m(\alpha)$ is a continuous function bounded at infinity, so is its norm, and the supremum of that function is finite, that is,

$$\sup_{\boldsymbol{\theta}} \| f(\boldsymbol{\theta}) \| = \sup_{\alpha} \| m(\alpha) \| = S < \infty$$

Since the supremum of the infinity norm of f is S, this implies that f maps the convex, bounded set $\overline{B(\mathbf{0},S^{2n})}\subset\mathcal{R}^{2n}$ into itself. Now, since $f(\boldsymbol{\theta})$ is a continuous function that maps a convex compact set into a convex compact set, then as a consequence of the Schauder Fixed Point theorems [16], there exists a point $\bar{\boldsymbol{\theta}}\in\overline{B(\mathbf{0},S^{2n})}$ such that $f(\bar{\boldsymbol{\theta}})=\bar{\boldsymbol{\theta}}$, *i.e.*, a fixed point exists.

3.4 Determination of $\mathbf{R}_{AB}$

The calculation of the $\mathbf{R}$ matrix is primarily dependent on the $\mathbf{T}_k$ matrix, but the $\mathbf{R}_{AB}$ matrix is also required. If the measurement noise is on A_k and B_k directly, then this is not a problem. However, to make it more complicated, direct measurements of A_k and B_k are generally not available. Rather, Fourier analyzer supplied measurements of magnitude and phase angle are available. Therefore in this section, three different scenarios are discussed: 1) Constant strength, uncorrelated noise on the real and imaginary phasor components directly; 2) Constant strength, uncorrelated noise on the magnitude and phase angle in radians as supplied by Matlab; and 3) Constant strength, uncorrelated noise on the magnitude in decibels and phase angle in degrees as supplied by a frequency analyzer.

3.4.1 Noise on the Phasor Components A_k and B_k. This is the most unlikely scenario of the three, but it is also the easiest and the one on which the derivation in Section 3.2 is based. Here, one assumes direct access to the real and imaginary parts of the frequency response data. This implies that the measurement noise strengths and correlation are known, *i.e.*, $\mathbf{R}_{AB}$ is known. A plot of a representative noise is shown in Fig. 3.2.

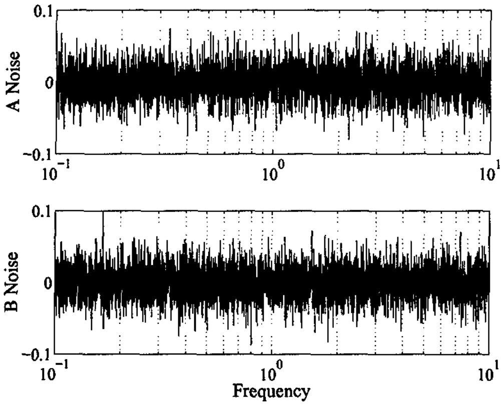


Figure 3.2. Constant Strength Uncorrelated Noise Added to A_k and B_k

3.4.2 Noise on the Matlab Supplied M_k and ϕ_k. As stated previously, direct measurements of A_k and B_k are generally not available. The more likely scenario is that one has hardware or software supplied measurements of magnitude (M_k) and phase angle (ϕ_k). The following equations are then used to calculate A_k and B_k.

$$A_k = M_k \cos\phi_k, \qquad B_k = M_k \sin\phi_k \tag{3.12}$$

It is more likely that the measurement noises ($\mathbf{v}_{M\phi}$) on the observables M_k ($\mathcal{N}(0, \sigma_M^2)$) and ϕ_k ($\mathcal{N}(0, \sigma_\phi^2)$) are uncorrelated, and this would lead to correlation between v_{A_k} and v_{B_k}. Representative noises for this scenario are shown in Fig. 3.3. As can be seen here, another complication is that the strength of the noise on A_k and B_k is different at each frequency. To remedy this, one can assume that the noises are small, and using Eq. (3.12),

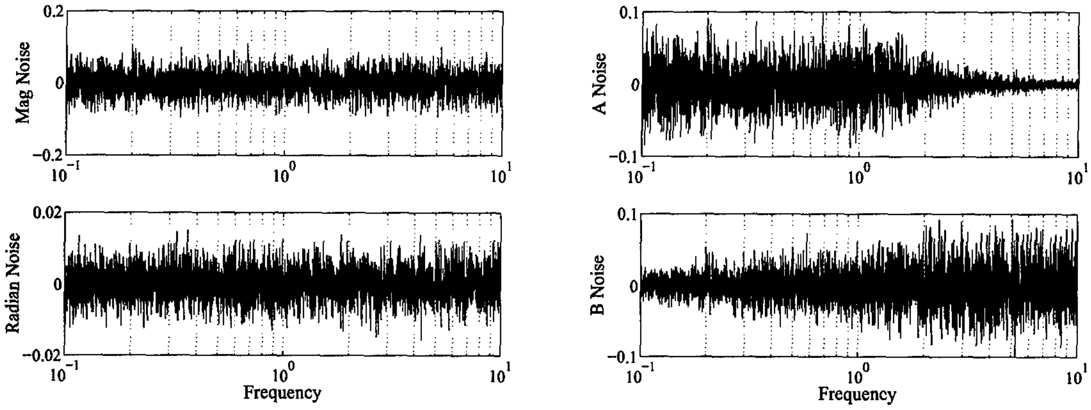


Figure 3.3. Constant Strength Noise on M_k and ϕ_k

the following linearized relation is derived.

$$\underbrace{\begin{bmatrix} \Delta A_k \\ \Delta B_k \end{bmatrix}}_{\mathbf{v}} = \underbrace{\begin{bmatrix} \cos\phi_k & -M_k \sin\phi_k \\ \sin\phi_k & M_k \cos\phi_k \end{bmatrix}}_{\mathbf{T}_{M\phi}} \underbrace{\begin{bmatrix} \Delta M_k \\ \Delta\phi_k \end{bmatrix}}_{\mathbf{v}_{M\phi}}$$

Now $\mathbf{R}_{AB}$ is calculated as follows.

$$\mathbf{R}_{AB} = E\left\{\mathbf{v}_k \mathbf{v}_k^T\right\} = \mathbf{T}_{M\phi} E\left\{\mathbf{v}_{M\phi} \mathbf{v}_{M\phi}^T\right\} \mathbf{T}_{M\phi}^T = \mathbf{T}_{M\phi} \mathbf{R}_{M\phi} \mathbf{T}_{M\phi}^T \tag{3.13}$$

This indicates that $\mathbf{R}_{AB}$ will be different for each measurement, and will need to be recalculated each time before adding $\mathbf{R}_k$ to the block diagonal $\mathbf{R}$ matrix. In practice, this calculation is not completely accurate, since the true values of M_k and ϕ_k, which are required for $\mathbf{T}_{M\phi}$, are not available. One can use the measured values in their place and achieve good results for smaller noise values, but much better results are obtained if one uses the parameter estimates to estimate the transformation matrix as well.

3.4.3 Noise on the Analyzer Supplied M_{kdB} and ϕ_{kdeg}. The motivation for this scenario arises from hardware and software considerations. Manufacturer's accuracy specifications for the Tektronix frequency analyzer [33] are given in dB and degree values, but the `cos()` and `sin()` functions in Matlab require radian values. This is handled through the derivation of an additional linearized transformation. The pertinent equations are

$$M_k = 10^{\frac{M_{kdB}}{20}} \qquad \phi_k = \phi_{kdeg}\frac{\pi}{180}$$

This leads to changes in $\mathbf{R}_{M\phi}$ at each frequency, as can be seen in Fig. 3.4, so once again assuming small values of noise, the following is derived.

$$\underbrace{\begin{bmatrix}\Delta M_k\\ \Delta\phi_k\end{bmatrix}}_{\mathbf{v}_{M\phi}}\underbrace{\begin{bmatrix}10^{\frac{M_{kdB}}{20}}\frac{\ln 10}{20} & 0\\ 0 & \frac{\pi}{180}\end{bmatrix}}_{\mathbf{T}_{dB^\circ}}\underbrace{\begin{bmatrix}\Delta M_{kdB}\\ \Delta\phi_{kdeg}\end{bmatrix}}_{\mathbf{v}_{dB^\circ}}$$

This implies that

$$\mathbf{R}_{M\phi} = E\left\{\mathbf{v}_{M\phi}\mathbf{v}_{M\phi}^T\right\} = \mathbf{T}_{dB^\circ}\mathbf{R}_{dB^\circ}\mathbf{T}_{dB^\circ}^T \tag{3.14}$$

so

$$\mathbf{R}_{AB} = \mathbf{T}_{M\phi}\mathbf{T}_{dB^\circ}\mathbf{R}_{dB^\circ}\mathbf{T}_{dB^\circ}^T\mathbf{T}_{M\phi}^T \quad \text{where typically} \quad \mathbf{R}_{dB^\circ} = \begin{bmatrix}\sigma_{M_{dB}}^2 & 0\\ 0 & \sigma_{\phi_{deg}}^2\end{bmatrix} \tag{3.15}$$

As before, the parameter estimates are used to estimate this transformation matrix as well. Additionally, in practice the $\mathbf{T}_{M\phi}$ and $\mathbf{T}_{dB^\circ}$ (and even $\mathbf{T}_k$) transformations can be combined into one transformation matrix for the calculation of $\mathbf{R}_k$.

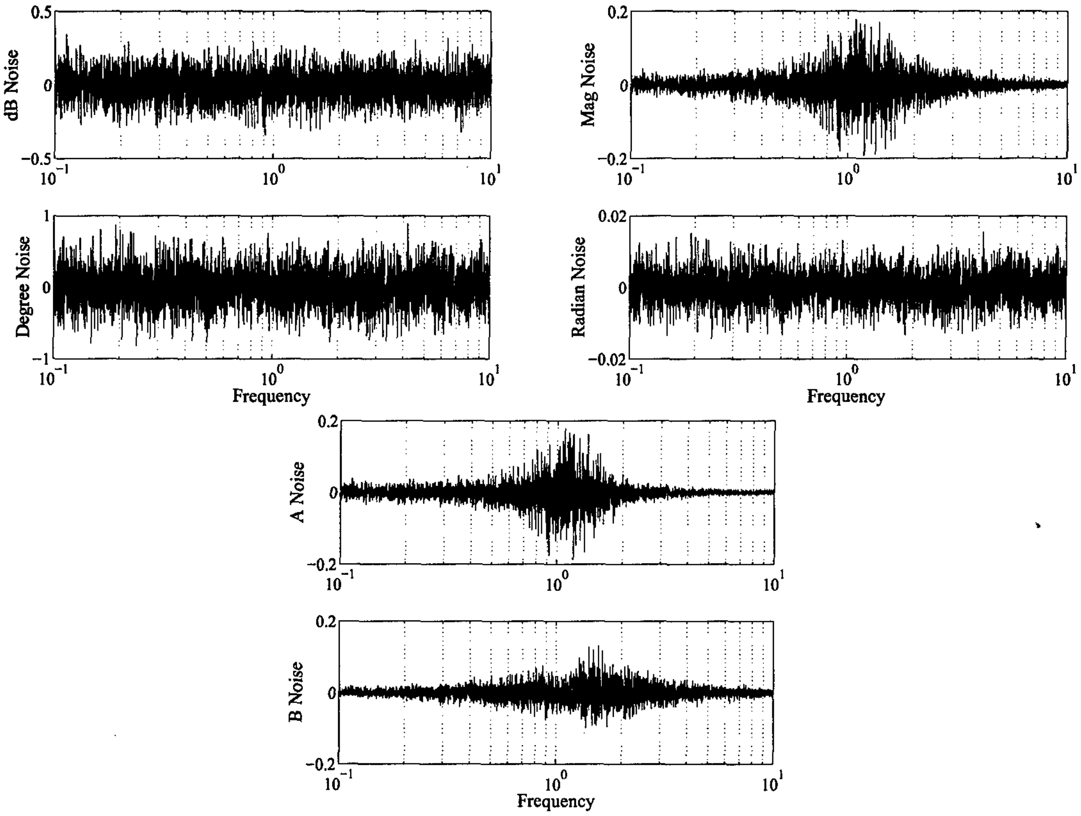


Figure 3.4. Constant Strength Noise on M_{kdB} and ϕ_{kdeg}

3.5 Other Methods of System Identification

As mentioned in Chapter II, there are a couple common methods of dealing with correlated system noises. The first method is the Instrumental Variable method. In this method, one calculates the next estimate using an $\widehat{\mathbf{H}}$ matrix which is the $\mathbf{H}$ matrix that would have resulted from a system consisting of the estimated parameters. In other words, the estimated parameters yield an estimated system. The elements of the $\widehat{\mathbf{H}}$ matrix are then the transformed magnitude and phase values off of the estimated Bode plot. The following equation is then iterated until a suitable convergence criterion is met:

$$\hat{\boldsymbol{\theta}}_{IV} = \left(\widehat{\mathbf{H}}^T\mathbf{H}\right)^{-1}\widehat{\mathbf{H}}^T\mathbf{z} \tag{3.16}$$

The second common method for dealing with correlated noise is the Generalized Least Squares method. A purported example of this is contained in Ref. [19]. The derivation of the noise model is essentially the same, but too many simplifying assumptions are made about the nature of the measurement noise. The final result can be shown to be identical to a Generalized Minimum Variance algorithm that assumes $\mathbf{R}_{AB} = \mathbf{I}$.

3.6 Second Order Example

For investigative purposes, a general second-order dynamical system, which is representative of an aircraft's elevator-to-pitch rate transfer function, is considered. The actual values are obtained from [2], and the transfer function is given by

$$T(s) = \frac{y(s)}{u(s)} = \frac{b_1 s + b_2}{s^2 - a_1 s - a_2} = \frac{4.8s + 1.44}{s^2 + 0.84s + 1.44} \tag{3.17}$$

The Bode plot for this transfer function, which is representative of the pitch dynamics of an aircraft, and which is used for inner-loop flight control system design, is shown in Fig. 3.5. Also shown are the forty measurement frequencies used in each of the following experiments.

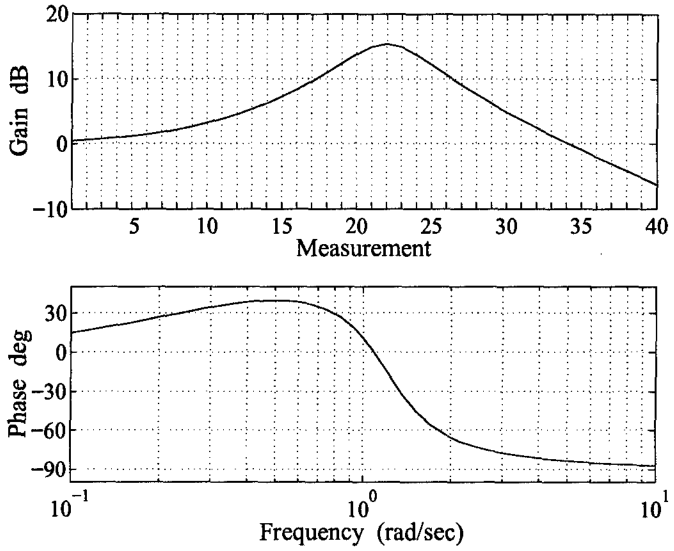


Figure 3.5. Second Order Bode Plot

The first thing that needs to be done is to obtain the proper noise model. For this second-order system, the equation error noise vector in Eq. (3.5) is given by

$$\boldsymbol{\epsilon}_k = \begin{bmatrix} -\frac{a_1 v_{A_k}}{\omega_k} - \frac{a_2 v_{B_k}}{\omega_k^2} - v_{B_k} \\ -\frac{a_1 v_{B_k}}{\omega_k} + \frac{a_2 v_{A_k}}{\omega_k^2} + v_{A_k} \end{bmatrix} = \begin{bmatrix} -\frac{a_1}{\omega_k} & -1 - \frac{a_2}{\omega_k^2} \\ 1 + \frac{a_2}{\omega_k^2} & -\frac{a_1}{\omega_k} \end{bmatrix} \begin{bmatrix} v_{A_k} \\ v_{B_k} \end{bmatrix}$$

This yields the 2 × 2 covariance matrix $\mathbf{R}_k$

$$\mathbf{R}_k = \begin{bmatrix} -\frac{a_1}{\omega_k} & -1 - \frac{a_2}{\omega_k^2} \\ 1 + \frac{a_2}{\omega_k^2} & -\frac{a_1}{\omega_k} \end{bmatrix} \begin{bmatrix} \sigma_A^2 & \sigma_{AB} \\ \sigma_{AB} & \sigma_B^2 \end{bmatrix} \begin{bmatrix} -\frac{a_1}{\omega_k} & -1 - \frac{a_2}{\omega_k^2} \\ 1 + \frac{a_2}{\omega_k^2} & -\frac{a_1}{\omega_k} \end{bmatrix}^T \tag{3.18}$$

whose elements are

$$\begin{aligned} \mathbf{R}_{k11} &= \sigma_A^2 \left(\frac{a_1^2}{\omega_k^2}\right) + \sigma_{AB}^2 \left(\frac{2a_1a_2}{\omega_k^3} + \frac{2a_1}{\omega_k}\right) + \sigma_B^2 \left(\frac{a_2^2}{\omega_k^4} + \frac{2a_2}{\omega_k^2} + 1\right) \\ \mathbf{R}_{k22} &= \sigma_B^2 \left(\frac{a_1^2}{\omega_k^2}\right) + \sigma_{AB}^2 \left(-\frac{2a_1a_2}{\omega_k^3} - \frac{2a_1}{\omega_k}\right) + \sigma_A^2 \left(\frac{a_2^2}{\omega_k^4} + \frac{2a_2}{\omega_k^2} + 1\right) \\ \mathbf{R}_{k12} &= \sigma_A^2 \left(-\frac{a_1}{\omega_k} - \frac{a_1a_2}{\omega_k^3}\right) + \sigma_{AB}^2 \left(\frac{a_1^2}{\omega_k^2} - \frac{a_2^2}{\omega_k^4} - \frac{2a_2}{\omega_k^2} - 1\right) + \sigma_B^2 \left(\frac{a_1}{\omega_k} + \frac{a_1a_2}{\omega_k^3}\right) \\ &= \mathbf{R}_{k21} \end{aligned}$$

As can be seen, if $\sigma_A^2 = \sigma_B^2 = \sigma^2$, and if v_A and v_B are uncorrelated, then $\mathbf{R}_k$ reduces to $\sigma^2\mathbf{I}_{2\times2}$. However, one still cannot use Eq. (2.2) to estimate $\boldsymbol{\theta}$ because $\mathbf{R}_k \neq \mathbf{R}_j$ when $\omega_k \neq \omega_j$.

3.6.1 Basis for Choice of Noise Strengths. During the course of this example, the three scenarios discussed in Section 3.4 are examined. An effort is made to keep the noise comparable in each of the sections. The noise strengths used in this example are representative of a Tektronix frequency analyzer. The manufacturer's specifications give measurement error values of ±0.2 *dB* and ±0.5 *deg*. These are taken conservatively as two sigma values for the noise on the amplitude and phase measurements.

For the constant *dB* and *deg* addition in scenario 2, a noise with a covariance matrix as shown in Eq. (3.15) with $\sigma^2_{M_{dB}} = (0.2/2)^2$ and $\sigma^2_{\phi_{deg}} = (0.5/2)^2$ is generated and added to the true *dB* and degree measurements obtained from the system shown in Fig. 3.5. These noisy *dB* magnitude and phase values are then converted to noisy amplitude and phase values, and finally to noise-corrupted A and B values. Each of the noisy values is then subtracted from its true value to obtain the transformed noise. The results of this addition and transformation process are shown in Fig. 3.4.

In Section 3.6.2, constant strength noise additions to A_k and B_k are examined. To determine a reasonable value for the noise strength in this experiment, the actual covariance of the A_k and B_k noise in Fig. 3.4 is calculated. The resulting covariance matrix is not a scaled version of the identity matrix, but for this experiment, an average of the diagonal terms is used for σ_A^2 and σ_B^2, and the off-diagonal terms are set to zero. This results in noise as shown in Fig. 3.2 with a covariance matrix of

$$\mathbf{R}_{AB} = \begin{bmatrix} 0.023^2 & 0 \\ 0 & 0.023^2 \end{bmatrix} \tag{3.19}$$

Section 3.6.3 discusses the identification results as the measurement noise strength is increased by powers of 10, *i.e.*,

$$\mathbf{R}_{AB_{new}} = 10^p * \mathbf{R}_{AB} \qquad p = 0, 1, 2, 3, 4 \tag{3.20}$$

This higher noise evaluation is also performed for the other noise scenarios discussed in Section 3.4.

In Section 3.6.5, a constant strength noise is added to the amplitude and phase variables. To determine appropriate strengths for these noises, the covariances of the M_k and ϕ_k noises shown in Fig. 3.4 are calculated, and these covariances are used to form constant strength noise additions as shown in Fig. 3.3. The resulting covariance matrix is

$$\mathbf{R}_{M\phi} = \begin{bmatrix} 0.0317^2 & 0 \\ 0 & 0.0044^2 \end{bmatrix} \tag{3.21}$$

3.6.2 Inadequacy of the Least Squares Method. In this section, one pretends that A_k and B_k are directly measurable. To compare the different estimation methods, uncorrelated and equal strength noises are added to the true A_k and B_k after they are computed from a Bode analysis of the transfer function in Eq. (3.17). Representative noise is shown in Fig. 3.2 with the resulting values of $\sigma_A = \sigma_B = 0.023$. This choice of $\mathbf{R}_{AB}$ results in a diagonal $\mathbf{R}$ matrix, but it is still not a scalar multiple of the identity matrix because of the varying measurement frequencies.

Figure 3.6 displays the results of a 100 run Monte-Carlo (MC) analysis, which upon evoking the weak law of large numbers [11], renders a gauge of the identification algorithm's estimation bias. The estimation results are first normalized by dividing each estimate by the true estimate, and then plotted. Ellipses are plotted representing each estimation method's actual one sigma variation. The ellipses' axes intersect at the average estimate for each method. The algorithm predicted estimation error covariances are not plotted, because they are not known for the LS or GLS methods, *i.e.*, Eq. (2.2) cannot be calculated because there is no constant σ that can be used. Indeed, the σ required in Eq. (2.2) is estimated as follows:

$$\hat{\sigma} = \sqrt{\frac{\tilde{\mathbf{z}}^T\tilde{\mathbf{z}}}{m-4}}\ , \qquad \tilde{\mathbf{z}} = \mathbf{z} - \mathbf{H}\hat{\boldsymbol{\theta}}$$

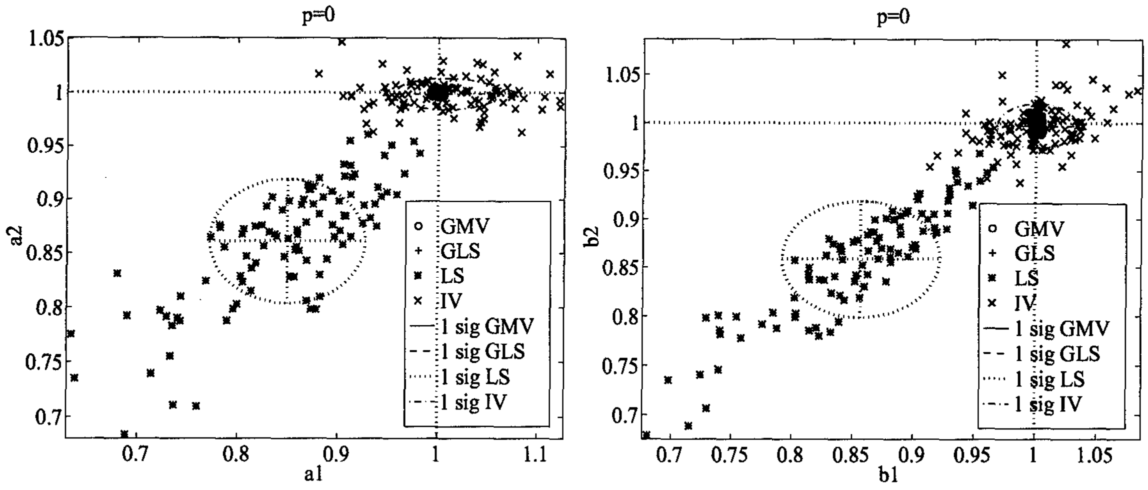


Figure 3.6. All Estimates with Uncorrelated A_k and B_k Noise

However, this estimate is data driven and the predicted estimation error covariance still would not be an accurate representation of the true error covariance. The available algorithm predicted covariances, along with all other numerical results, are given in Table B.1.

As can be seen in the plots, there are large biases in the LS average estimate, and the true parameter is well outside the one sigma bounds. This indicates, as is shown in the plot, that the majority of the parameter estimates in the hundred runs are further from the true estimate than one sigma.

Dramatic improvements can be made by using one of the other estimation methods. As seen in the plot, the IV method does have quite a large sigma value, but the estimate is not nearly so biased as the LS. The GLS and GMV estimates cannot be distinguished on this plot because they are tightly clustered around the true estimate. Figure 3.7 zooms in to give a better look at these estimates. This magnification provides a good view of the IV estimates and sigma, but the GLS and GMV estimates are still clustered rather tightly around the true parameter. Indeed, this is always the case. The GLS and GMV estimates are always better than the LS and IV. For this reason, the LS and IV estimates are dropped from most plots in the sequel. The numerical results are still included in the appendix.

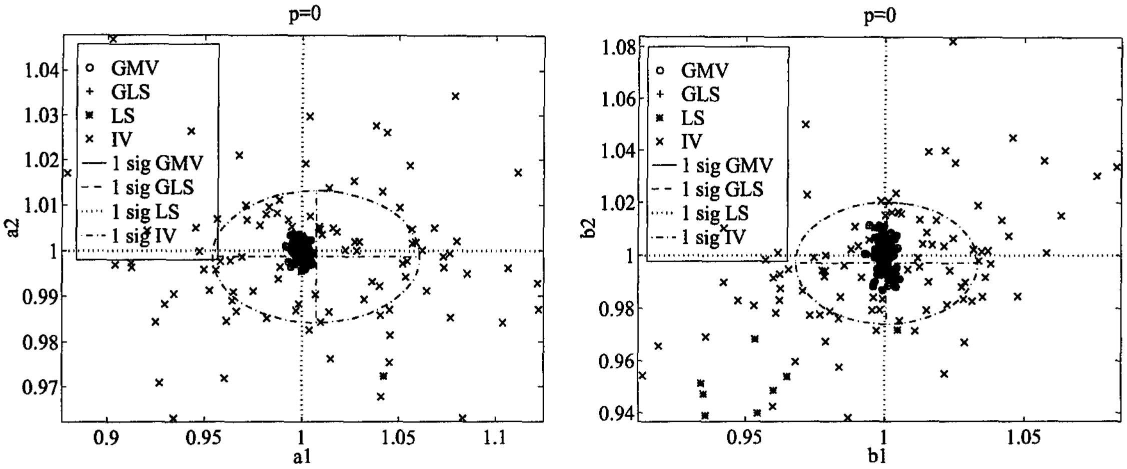


Figure 3.7. Magnification of Estimates with Uncorrelated A_k and B_k Noise

Further magnification in this noise scenario would provide little information on the comparison of the GLS and GMV estimates because they are identical in this case. The only difference between the two is a scaling of the **R** matrix which cancels out in the final equation. The only thing gained by the GMV method is an accurate estimation of the estimation error covariance. As is seen in Table B.1, these covariance estimates are rather close to the actual ones. The small biases and covariance differences imply that the GMV and GLS system identification algorithms yield relatively unbiased parameter estimates. Additionally, GMV is doing a good job of predicting the accuracy of its estimate, *i.e.*, the algorithm is "efficient."

3.6.3 Increasing Noise. The next portion of the experiment involves increasing the level of the measurement noise, as in Eq. (3.20). The results for $p = 1$ are very similar to $p = 0$. But when p is increased to 2, a trend that causes problems later on starts to appear. As the strength of the measurement noise increases, the LS estimates start to cluster around the origin (Fig. 3.8). This may seem like it should not effect the GMV estimate, and it does not in this case, but recall that the GMV estimate is initialized with this LS estimate. When p is increased to 3, the GMV estimates also start to cluster around **0**. This is not a

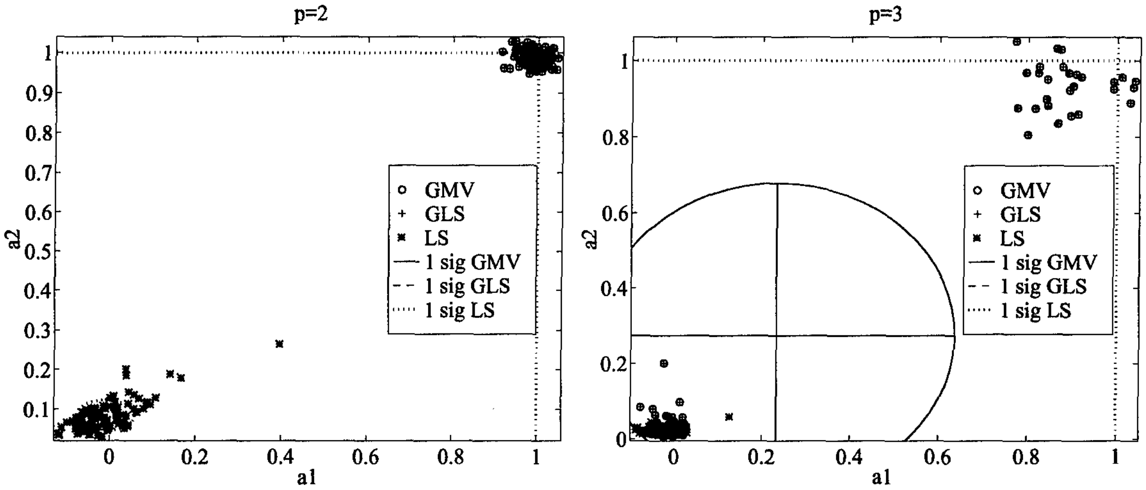


Figure 3.8. Higher Noise Estimates with Uncorrelated A_k and B_k Noise

gradual migration like the one that happens in the LS estimates as the noise increases; it is sudden. Indeed, it is caused by the appearance of a second fixed point close to the origin.

To better illustrate this problem, a "shotgun approach" is taken for one particular noise realization. The results of the GMV algorithm, when it is initialized with a series of different vectors whose elements range from -0.5 to 0.5, are shown in Figure 3.9. The left figure shows the paths taken during the iterations. The first iteration step is shown by a dotted line, while the remaining iterations are plotted as a solid line. The end points (or fixed points) are shown as an *. Most of the initial value choices converge to the desired estimate, but several runs initialized close to the origin converge to a different value. This seems to be a detrimental property of the phasor oriented GMV algorithm at high noise. How does one choose the "correct" fixed point? The solution to this may lie in the right hand plot. This is a plot of $[f(\boldsymbol{\theta}) - \boldsymbol{\theta}]^T[f(\boldsymbol{\theta}) - \boldsymbol{\theta}]$ at each iteration. The slowly converging paths in this plot correspond to those paths that are converging to the wrong point. Although the LS estimate is a convenient point to start the GMV algorithm, it is not the wisest choice, since it tends to approach the origin as the noise level increases.

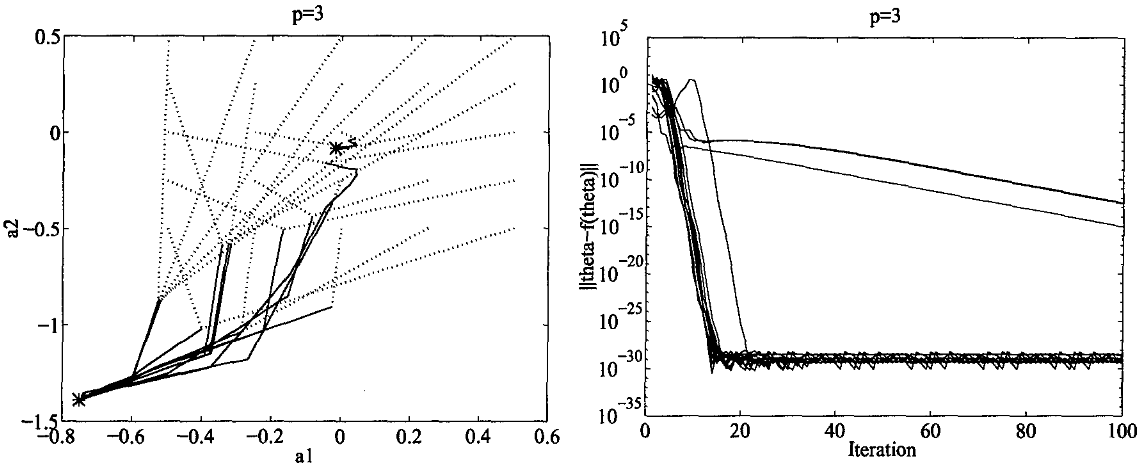


Figure 3.9. Convergence Properties for Uncorrelated A_k and B_k Noise

A better choice would be a point in the same quadrant as the true parameter, but with a larger magnitude. This is usually not possible, because one does not know the true parameter. However, if the problem is known to be bothered by high noise, one can choose several starting points, perhaps one from each quadrant, and the one that converges the fastest is the correct one.

If one examines the convergence rates for the $p = 3$ estimates in Fig. 3.8, which are shown in Fig. 3.10, it can be seen that many estimates converge very slowly, while others do not appear to converge at all. However, when the GMV algorithm is initialized at the point $\boldsymbol{\theta}_0 = [-100\ \ -100\ 0\ 0]^T$, the results improve dramatically. These results are shown in Fig. 3.11. As can be seen, the GMV estimates are no longer trapped around the origin, and the convergence rates are much quicker as well.

3.6.4 Noise on Observables. In this section, the noise is added to the *dB* magnitude and phase angle (in degrees), as is given in the Tektronix [33] specifications. Then, the two required transformations (Eqs. (3.13) and (3.15)) are performed to obtain the correct σ_A^2, σ_B^2, and σ_{AB}^2 used for the $\mathbf{R}_k$ matrix calculation in Eq. (3.18). Through proper modeling of the noise transformations, the GMV algorithm achieves better estimates than any of the

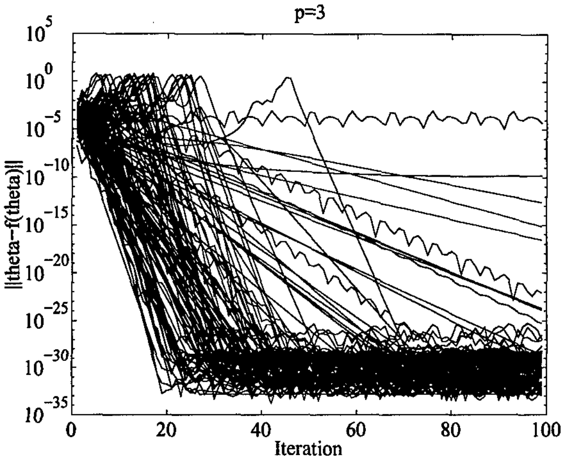


Figure 3.10. Convergence Rates for LS Initialization

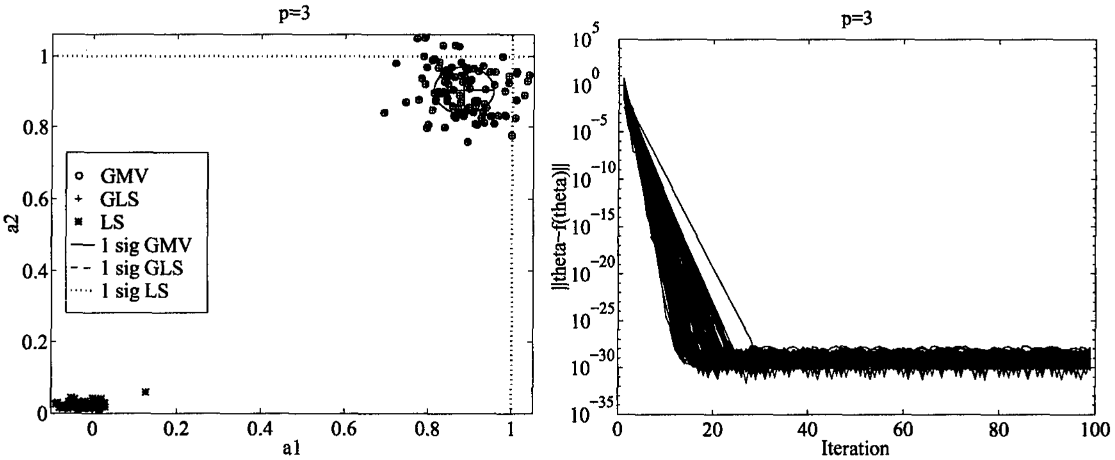


Figure 3.11. Estimates and Convergence Rates for Modified Initialization

other methods. The GLS algorithm comes the closest, and a comparison of these for the $p = 0$ case is shown in Fig. 3.12.

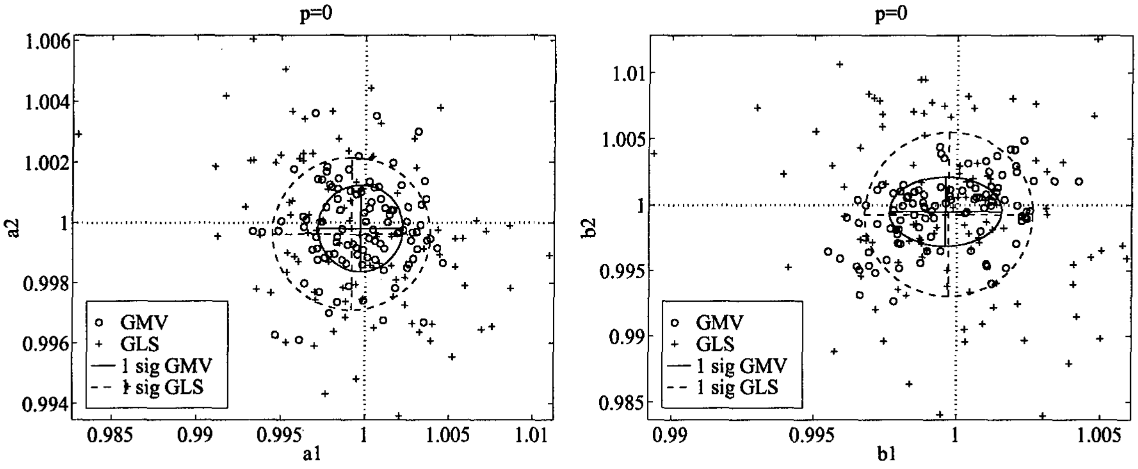


Figure 3.12. Estimates with Uncorrelated M_{kdB} and ϕ_{deg} Noise

As can be seen, the GMV estimates have both a lower bias and error covariance than the GLS estimates. The reason for this is the proper modeling of the $\mathbf{R}_{AB}$ matrix. Initial attempts at this used the measured values in the transformation matrices. While this yields good results for small noise strengths, the measured values for higher noise strengths do not provide an accurate estimate of the magnitude and phase required for the transformation matrix calculations. Problems arise in poor numerator estimates, and the parameters start to get trapped around the origin sooner than they would otherwise.

Instead, the measured values are used only for the first five GMV iterations. This is only done to get a more accurate estimate of the Bode plot, and may not even be necessary. After the fifth iteration, the GMV parameter estimates are used to estimate the Bode plots, and the magnitude and phase values from this estimation are used in the transformation matrices. This method produces much more accurate estimates, and it is also more robust to the "zero trapping" than before.

Higher noise strength runs were also performed for this scenario, and the numerical results are given in Table B.3. As before, the GMV results are the best, and the error covariance prediction is close to the actual.

3.6.5 Noise on Amplitude and Phase. In this section, the noise on the amplitude and phase is assumed to be of constant strength and temporally uncorrelated. To this end, the covariance matrix in Eq. (3.21) is used to generate noises for addition to M_k and ϕ_k. This implies that only one transformation needs to be performed to determine σ_A^2, σ_B^2, and σ_{AB}^2. Representative noises for this section are shown in Fig. 3.3. Once again, the measured values are used in $\mathbf{T}_{M\phi}$ for the first five GMV iterations, and then the parameter estimates are used to estimate $\mathbf{T}_{M\phi}$. The numerical results for all the noise strength cases are given in Table B.2.

One concern of note in this scenario is shown in Fig. 3.13. This is a plot of the error mean and covariance for the GMV algorithm as each measurement is added. The algorithm estimated covariance is close to the actual covariance after about 22 measurements, but the actual covariance appears to begin to diverge toward the end. The 22 measurements is not of concern, because this is highly dependent on the order in which the measurements are added to the linear regression. However, the cause of the divergence is not known. It has something to do with the $\mathbf{T}_{M\phi}$ transformation, because the covariance increase does not appear in the GLS estimates. A possible cause is the negative measured magnitudes at this noise level in this scenario. This may violate some physical property or assumption, but a definitive cause can not be determined.

3.7 Discussion

The normalized numerical results of all the experiments are summarized in Tables B.1 through B.3 in Appendix B. In all cases, the values shown correspond to a forty measurement linear regression. The Monte-Carlo averaged estimation error (e) and sigma (σ_e) are given for all cases, and the algorithm predicted sigma (σ_p) is given for the MV and

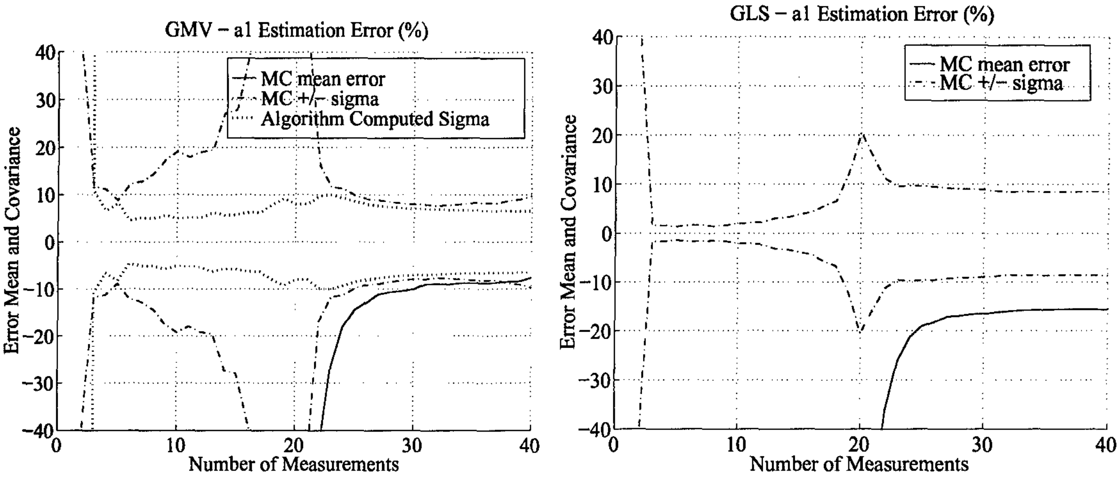


Figure 3.13. Estimation Performance for Increasing Measurements

IV experiments. In each case, the bias in the MV estimate is about two orders of magnitude smaller than in the naive LS estimate, and the estimation error covariance is much smaller as well.

For $p < 2$, all 100 of the estimated Bode plots are virtually indistinguishable. Appendix B contains the final estimation results for $p = 2, 3$. The top plots show the true Bode plot and the average of the estimated Bode plots, along with the range of the estimated Bode plots. As can be seen, the spread is quite small even for $p = 2$. It appears, however, that the admittedly small, identification provided, uncertainty is well within the system's bandwidth, *i.e.*, it is *structured* uncertainty.

The bottom plots show the true poles and zero, along with the estimated poles and zeros. For $p < 2$, the estimated poles lie practically on a horizontal line, and even at $p = 2$, the vertical variation is very small, *i.e.*, the identification algorithm renders a very accurate estimate of the damped natural frequency. A noticeable bias begins to appear when p reaches 3, but it is not all that large, considering the noise level.

3.8 Conclusions

In this chapter, a frequency domain approach is taken, and phasors are used for system identification. Gaussian measurement noise is assumed, as is customary in classical filtering and system identification work. The proper minimum variance estimate equations are derived and applied to a nonlinear estimation problem. The results are then compared to the simple minded least squares estimate for a second-order system that is representative of an aircraft's pitch dynamics, which is used for inner-loop flight control system design.

The GMV estimate outperformed the LS estimate in *all* cases. The LS estimate did, however, provide a useful, albeit dangerous, starting point for iterating the GMV estimate. At small noise levels, it does not matter where the GMV algorithm is initialized, but as higher noise levels appear, it is dangerous to initialize the algorithm close to the **0** point. There is an additional fixed point appearance there that has proven capable of trapping the estimate. A positive note, when the estimate is trapped, the algorithm takes much longer to converge, so multiple high magnitude starting points should allow one to find the desired fixed point. In conclusion, the least squares estimate is not an effective one, even in cases of small measurement noise. Although the IV estimate is somewhat better than the LS estimate, it still performs poorly under increased noise levels. The GLS and GMV provide much more accurate estimates, and the GMV seems to provide lower biased and smaller covariance estimates in each case.

In this chapter, it is shown that the frequency domain can be used in system identification. Careful stochastic modeling of the estimation problem at hand renders an efficient identification algorithm that is superior to straightforward least squares. However, the experimental phasor approach is applicable to stable systems only.

IV. Effects of Unmodeled Dynamics on the Phasor Approach

When system identification is performed for control system design, it is oftentimes required to identify the plant's dominant mode only. Thus, identification is performed in the presence of additive modes, and one then refers to unmodeled dynamics. In this chapter, the proper choice of identification signals when unmodeled dynamics are present is investigated. Different combinations of high and low frequency dynamics are added to a representative and physically motivated second-order plant, and carefully controlled identification experiments are performed. Moreover, in adding additional dynamics, one is strongly guided by physical considerations, and this results in a well-designed experiment. To solidify the concept, consider the flight control context.

Flight control system designers often use a plant model which consists of the short period pitch dynamics of an aircraft. However, the short period dynamics are not the only modes present in the aircraft's pitch channel. This directly impacts system identification. Thus, it is desired to identify the dominant mode of a system when there are other unmodeled dynamics present as well. Now, the more one knows about a particular system *a priori*, the easier the identification problem becomes. Prior information, including bandwidth information, is at a premium in system ID. Hence, the focus in this chapter is on the interaction between the unmodeled dynamics and the proper choice of input frequencies used to identify a plant.

In Section 4.1, a simple second-order plant representative of the short period dynamics of an aircraft, and no additional dynamics, is used to establish a baseline for determining optimal excitation frequencies. The short period dynamics are routinely used in flight control design work. Low and high frequency modes are added to the second-order plant in Sections 4.2 and 4.3, respectively. The idea here is that modeling based on physical considerations renders mathematically tractable problems. Hence, the low frequency mode is chosen to represent the aircraft's phugoid mode, and the high frequency dynamics model a flexible mode. Both the low and high frequency modes are included in the analysis and experiments presented in Section 4.4, and the results of naively overmodeling a system are in Section 4.5.

4.1 Second-Order Plant

In this section, the proper choice of frequencies used to identify a second-order underdamped plant, with one zero, is examined. This constitutes a baseline for comparing with results obtained in the presence of unmodeled dynamics. For this purpose, the plant given in Eq. (4.1), which is representative of the short period dynamics of an aircraft, is used. The Bode plot of its transfer function is shown in Fig. 4.1, from which the plant's bandwidth is readily apparent.

$$T_2(s) = \frac{4.8s + 1.44}{s^2 + 0.84s + 1.44} = \frac{4.8(s + 0.3)}{s^2 + 2(0.35)(1.2)s + (1.2)^2} \tag{4.1}$$

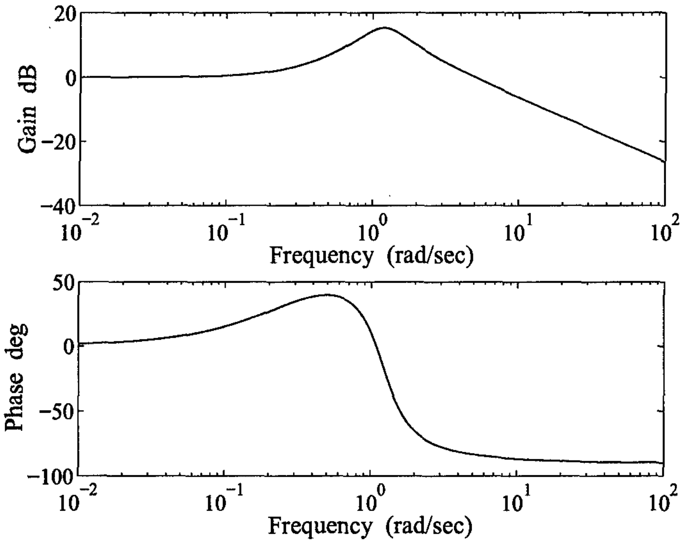


Figure 4.1. Second-Order Bode Plot

To examine the impact on identification of the choice of input frequency, the transfer function in Eq. (4.1) is used to generate a series of measurements, and Gaussian noise of strength $\sigma_{M_{dB}} = 0.1$ and $\sigma_{\phi_{deg}} = 0.25$ is then added to the measurements. The frequency range shown in Fig. 4.1 is then divided to provide 16 measurements per half decade. Finally, a half decade window is stepped through the given frequency range at 1/4 decade increments, resulting in 15 different parameter estimation frequency windows. For each window, twenty Monte-Carlo runs are performed using the estimation algorithm described in the previous chapter. These results are graphically shown in Fig. 4.2, and the actual values are summarized in Table C.1.

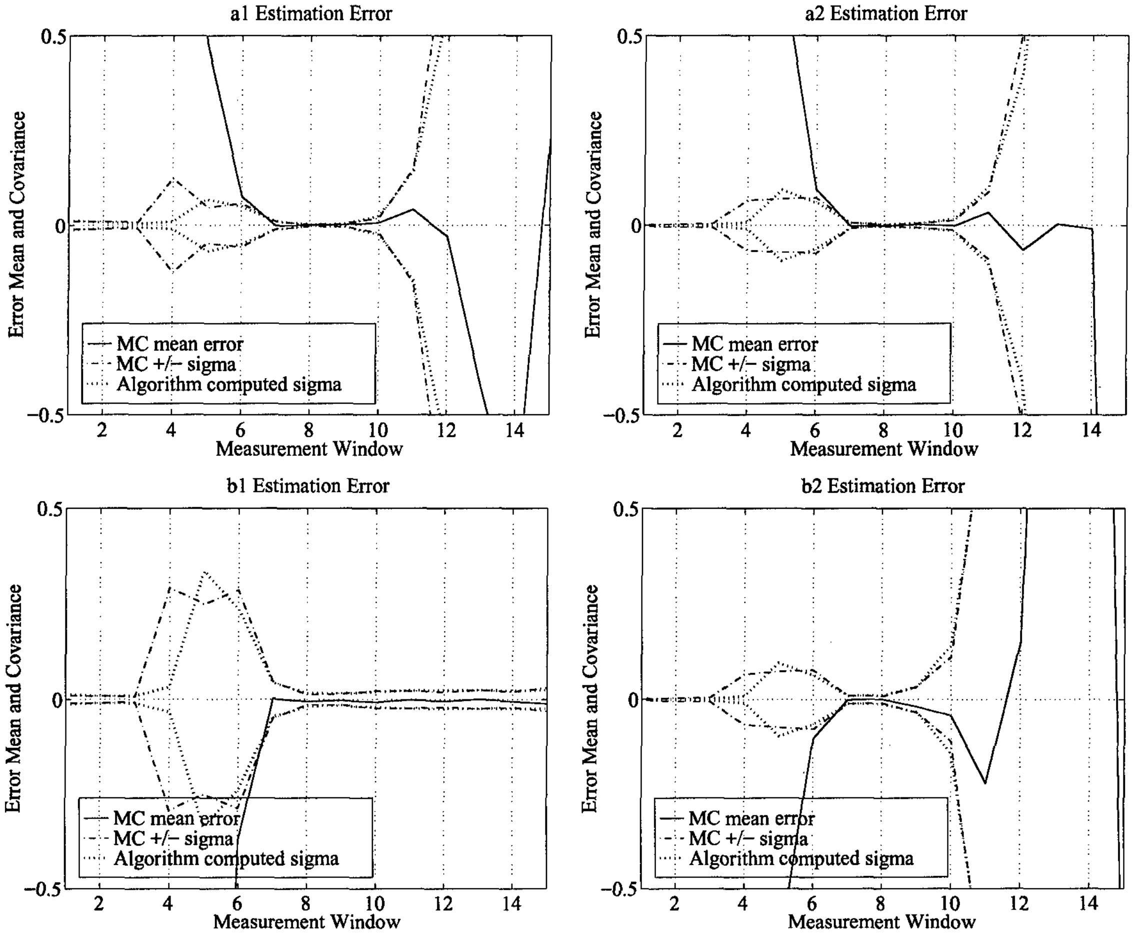


Figure 4.2. Results of Identifying the Second-Order Plant

As can be seen, the bias in the estimate is extremely large when very low frequencies are used for identification. An interesting note is that the covariance is small. This is truly poor ID because the algorithm believes that its estimate is accurate, when in fact the opposite is true. In the parlance of Kalman Filtering, one refers to filter divergence – an unfortunate manifestation of nonlinearities encountered when extended Kalman filters are used. Furthermore, the algorithm is identifying the plant as a first-order unstable plant, which may be caused by the effects of the transfer function zero at lower frequencies. The covariance then increases a bit as the bias approaches zero, and returns to small values around the center of the frequency range.

The estimation bias is smallest overall at window number seven, which corresponds to the half decade window just below 1 rad/sec. This window is one below that which contains the second-order mode's natural frequency. Also, the pole-zero estimation is most accurate at the same window. The estimation bias then becomes large again at the high frequencies, but the covariance increases as well. This is a better result than that yielded by the low frequency inputs because the algorithm is warning the user that its estimate is poor. The sole exception to the increasing bias is the estimate of b_1, which remains accurate. This seems to imply that the gain estimate may still be accurate.

The low and high measurement frequency problems, noted in this section for a second-order plant, form the basis for the analysis in the following sections. These sections include the addition of low and/or high frequency dynamics.

4.2 Low Frequency Unmodeled Dynamics

In this section, a second-order identification is again performed. However, the plant dynamics here include an unmodeled, low frequency, mode in addition to the short-period mode. Guided by insights into the physics of flight dynamics, this mode is similar to the phugoid mode of an aircraft [2]. Thus, the fourth-order elevator deflection to pitch rate transfer function is given by

$$T_{4l}(s) = \frac{4.7843s(s+0.016)(s+0.2862)}{(s^2+0.00466s+0.0053)(s^2+0.8394s+1.4383)} \tag{4.2}$$

To derive $T_{4l}(s)$, a particular low frequency mode is chosen and included in a fourth-order transfer function. Then, a numerical optimization routine is used to minimize the distance between the Bode plot of the fourth-order transfer function and the Bode plot of $T_2(s)$, at frequencies between $10^{-0.5}$ and $10^{0.5}$ rad/sec. The transfer function in Eq. (4.1) results. This match is performed in a noiseless environment, so the parameters in Eq. (4.1) are used to assess the error in the estimate obtained assuming a second-order model and using the fourth-order plant's measurements. The Bode plots of the plants in Eqs. (4.1) and (4.2) are compared in Fig. 4.3. As is seen, the Bode plots match around the natural frequency of the "dominant" mode and above, but differ dramatically around the natural frequency of the low frequency mode and below.

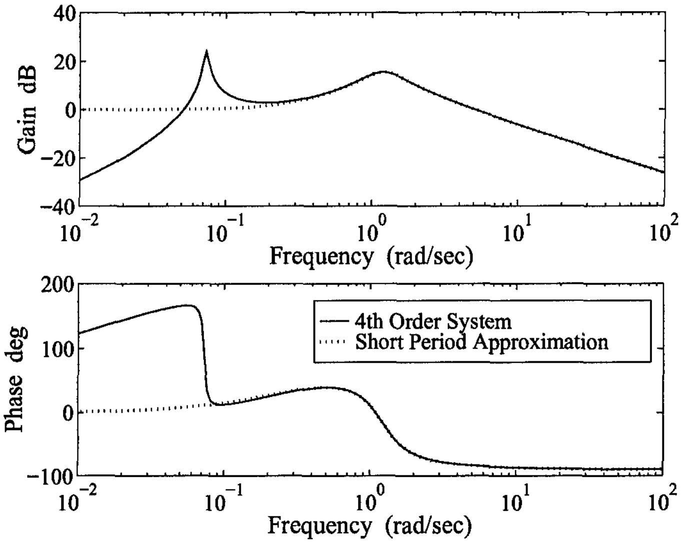


Figure 4.3. Second- and Fourth-Order Bode Plot Comparison

The same moving frequency window estimation is performed using the noise corrupted outputs of the fourth-order plant. The graphical results are portrayed in Fig. 4.4 and the actual numerical values are contained in Table C.2.

In this case, the low frequency estimation error covariances have collapsed to approximately zero, while the bias remains very large. The addition of the low frequency dynamics has hurt the estimation process. If one is naive enough to take low frequency measurements into account, the algorithm would produce a seemingly accurate estimate, when in fact it

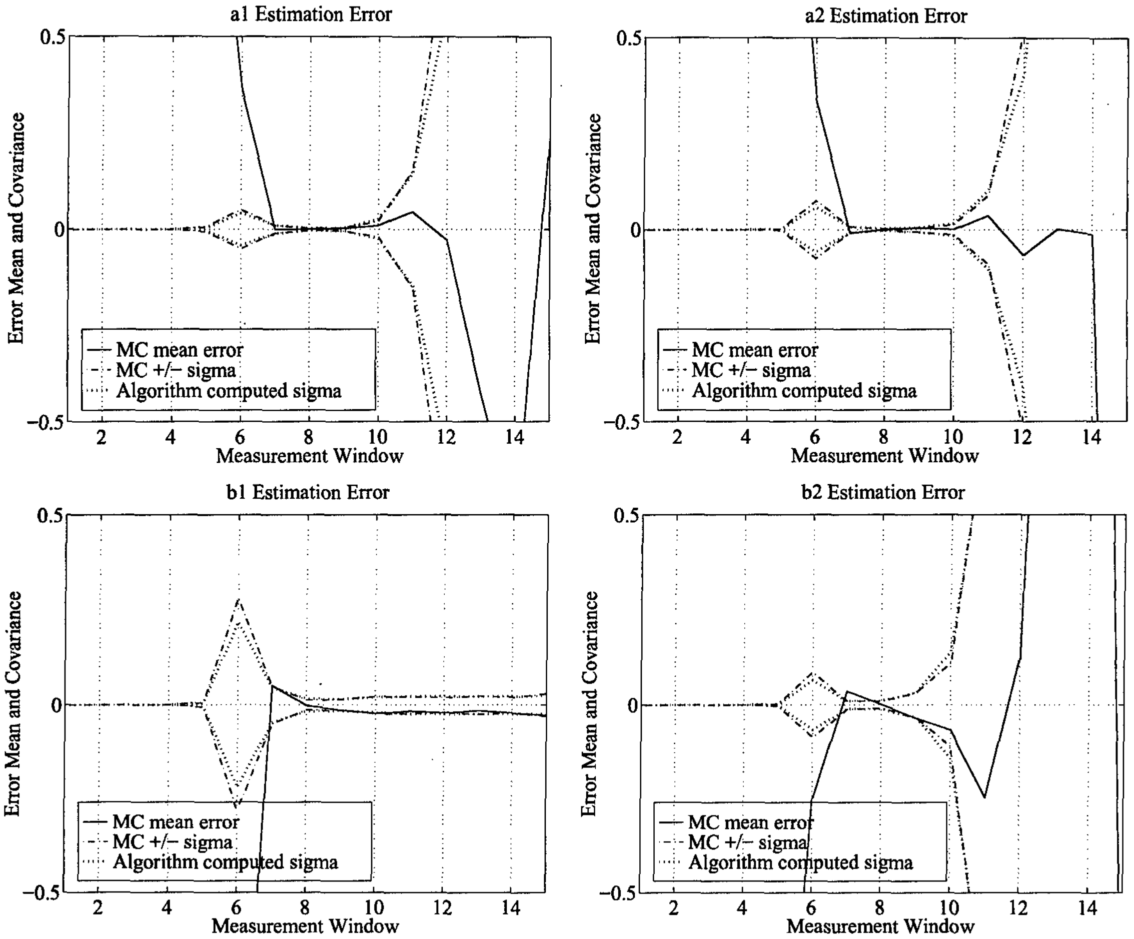


Figure 4.4. Results of Identifying the Fourth-Order Low Frequency Plant

is totally wrong. The error covariances then balloon as the estimation window moves into the range of the desired mode and collapse once again as the bias approaches zero. Additionally, the high frequency results are virtually identical to those of the previous section. An interesting note is that the most accurate estimate of the parameters, poles, and zero has now moved to window number eight; the one containing the desired dominant mode's natural frequency.

4.3 High Frequency Unmodeled Dynamics

In this section, the procedure is repeated with the low frequency mode replaced by a high frequency mode. Again, the problem is formulated by deliberately leaning toward reliance on physical insight into the problem. Thus, this mode is similar to the high frequency bending mode of an aircraft [26], and the fourth-order transfer function is given by

$$T_{4h}(s) = \frac{5.733(s+0.2979)(s^2+0.496s+695)}{(s^2+0.8394s+1.4383)(s^2+2.86s+827)} \tag{4.3}$$

Here a high frequency mode is chosen and included in a fourth-order transfer function. Then an optimization routine is once again used to minimize the distance between the Bode plot of the fourth-order transfer function and that of Eq. (4.1) at frequencies between $10^{-0.5}$ and $10^{0.5}$ rad/sec. Therefore as before, the parameters in Eq. (4.1) are deemed to be the desired parameters. The Bode plots of the plants in Eqs. (4.1) and (4.3) are compared in Fig. 4.5. As is seen, the Bode plots match at low frequencies, but differ in the neighborhood of the high frequency mode.

The same moving frequency window estimation is performed using the noise corrupted outputs of the fourth-order plant. The graphical results are portrayed in Fig. 4.6 and the actual numerical values are contained in Table C.3.

Once again, the additional mode introduces unmodeled dynamics, which potentially hamper the identification process. The low frequency performance is similar to the second-order results, but the high frequency error covariances collapse again in the area around the additional mode's natural frequency. Even the estimate of the b_1 parameter now has a large

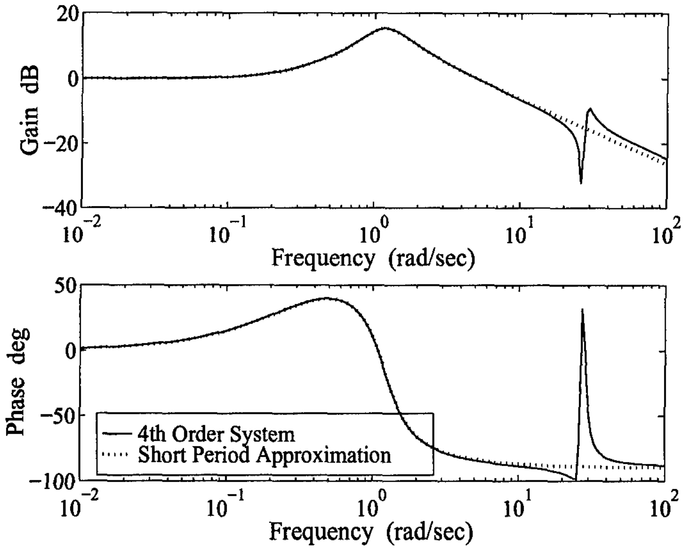


Figure 4.5. Second- and Fourth-Order Bode Plot Comparison

bias and small error covariance. The area of covariance collapse is not as large as when the low frequency mode is added, but as in the case of low frequency unmodeled dynamics, it corresponds to the area where the second-order Bode plot differs from the fourth-order.

The window of best estimation is also more ambiguous. Best parameter and pole-zero estimates range between windows eight and nine, depending on the parameter. This seems a bit strange when one considers the high frequency mode should push the better identification to lower frequencies. However, the fourth-order transfer function is matched to a second-order in a subset of window eight, which implies the best match should always be in window eight.

4.4 High and Low Frequency Unmodeled Dynamics

In this section, both the high frequency and low frequency modes are added to the plant, and the system identification procedure is repeated. This is similar to the situation in a real aircraft, and the sixth-order transfer function is given by

$$T_6(s) = \frac{5.7141s(s+0.016)(s+0.2841)(s^2+0.496s+695)}{(s^2+0.00466s+0.0053)(s^2+0.8424s+1.4405)(s^2+2.86s+827)} \tag{4.4}$$

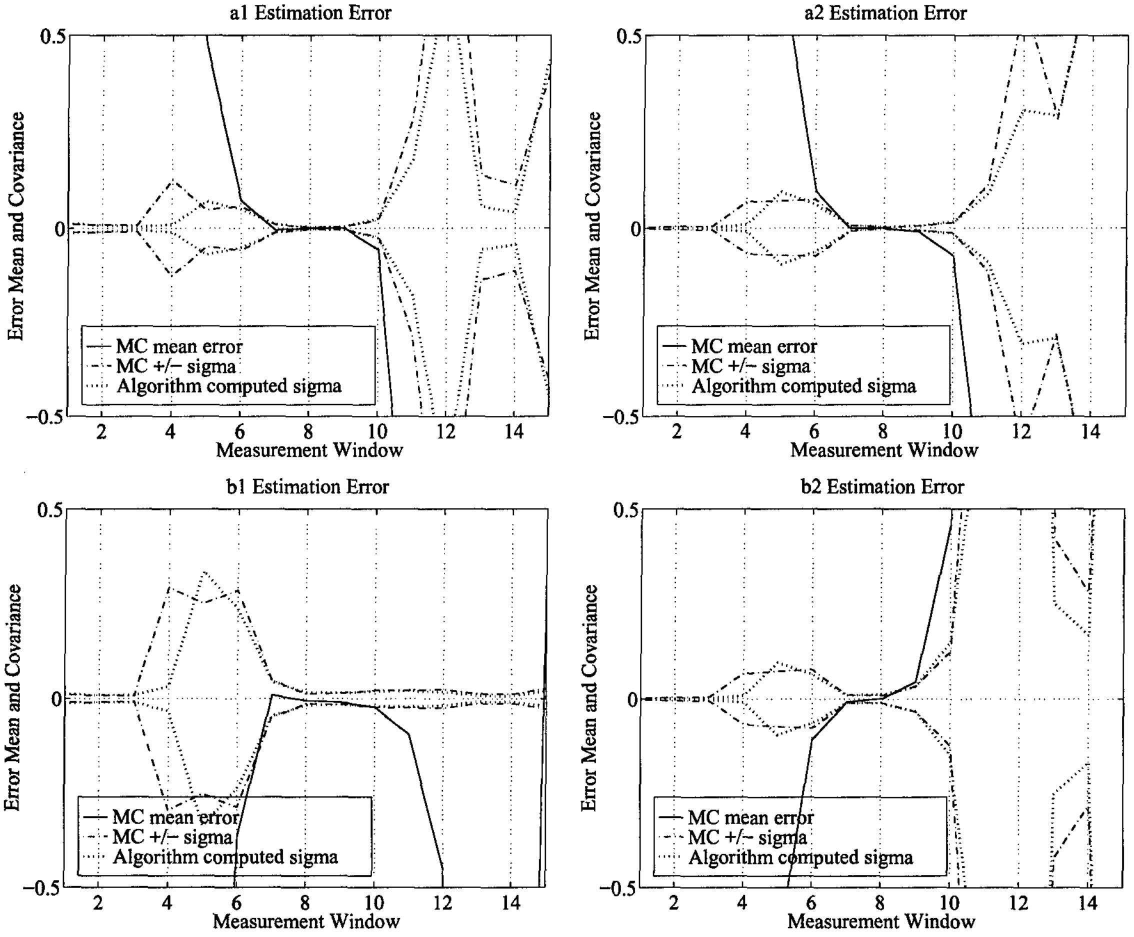


Figure 4.6. Results of Identifying the Fourth-Order High Frequency Plant

$T_6(s)$ is the result of using the same high and low frequency modes as before, and an optimization routine to match a sixth-order plant's Bode plot to that of Eq. (4.1) at frequencies between $10^{-0.5}$ and $10^{0.5}$ rad/sec. Therefore as before, the parameters of this second-order transfer function are used as the desired parameters. The Bode plots of the plants in Eqs. (4.1) and (4.4) are compared in Fig. 4.7. Obviously, the Bode plots now differ at both low and high frequencies.

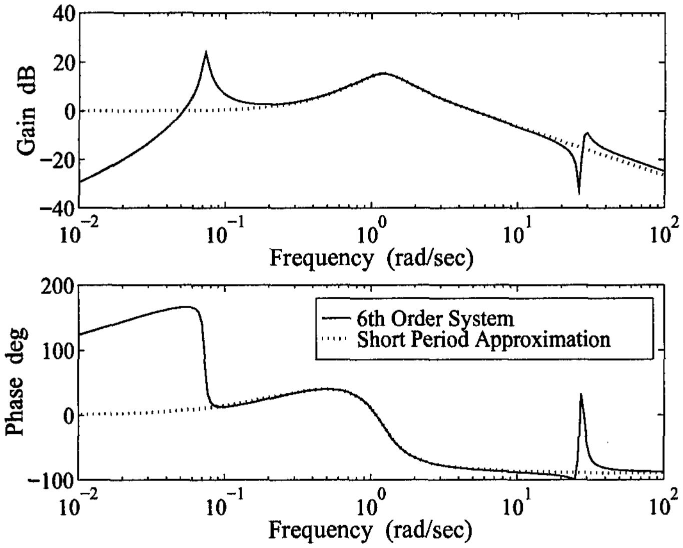


Figure 4.7. Second- and Sixth-Order Bode Plot Comparison

The same moving frequency window estimation is performed using the noise corrupted outputs of the sixth-order plant. The graphical results are portrayed in Fig. 4.8 and the actual numerical values are contained in Table C.4.

The estimation problems noted before are now compounded. The error covariance is now artificially small at both low and high frequencies. This collapse of estimation performance occurs when the Bode plots of the actual and desired plants differ greatly. Additionally, the best estimation is now definitely achieved in window eight. This further supports the notion that the best estimate occurs in the window where the Bode plots are matched.

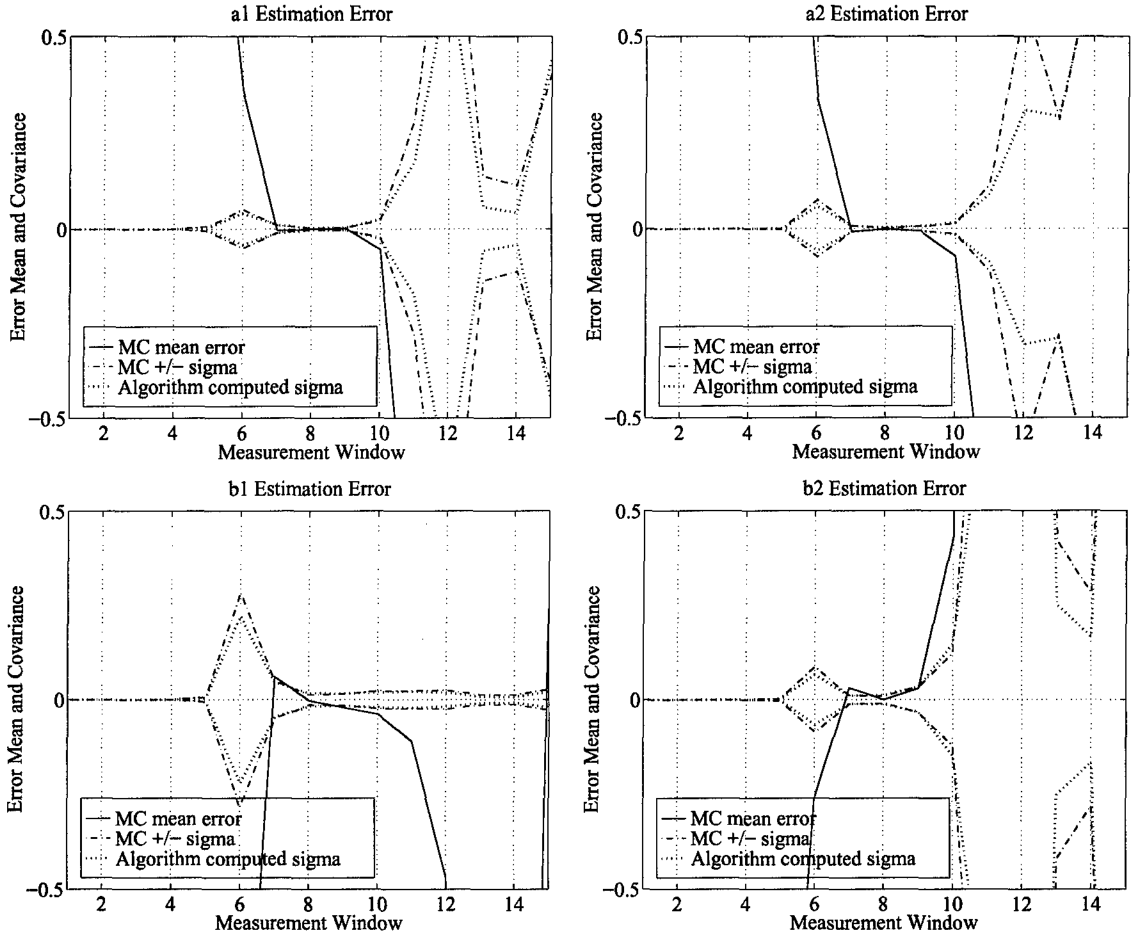


Figure 4.8. Results of Identifying the Sixth-Order Plant

4.5 Attempt at Overmodeling

In this section, the effect of overmodeling in the face of limited measurements is examined. The plant is modeled as sixth-order, but each estimate still only uses 16 measurements. As expected, the results are very poor, and representative samples are shown in Fig. 4.9. Compounding the problem that 16 measurements are not sufficient to identify 10 parameters accurately, is the fact that none of the frequency windows provides enough information about the full bandwidth for an accurate estimate. This emphasizes the idea of keeping the number of estimated parameters small as related to the measurement information.

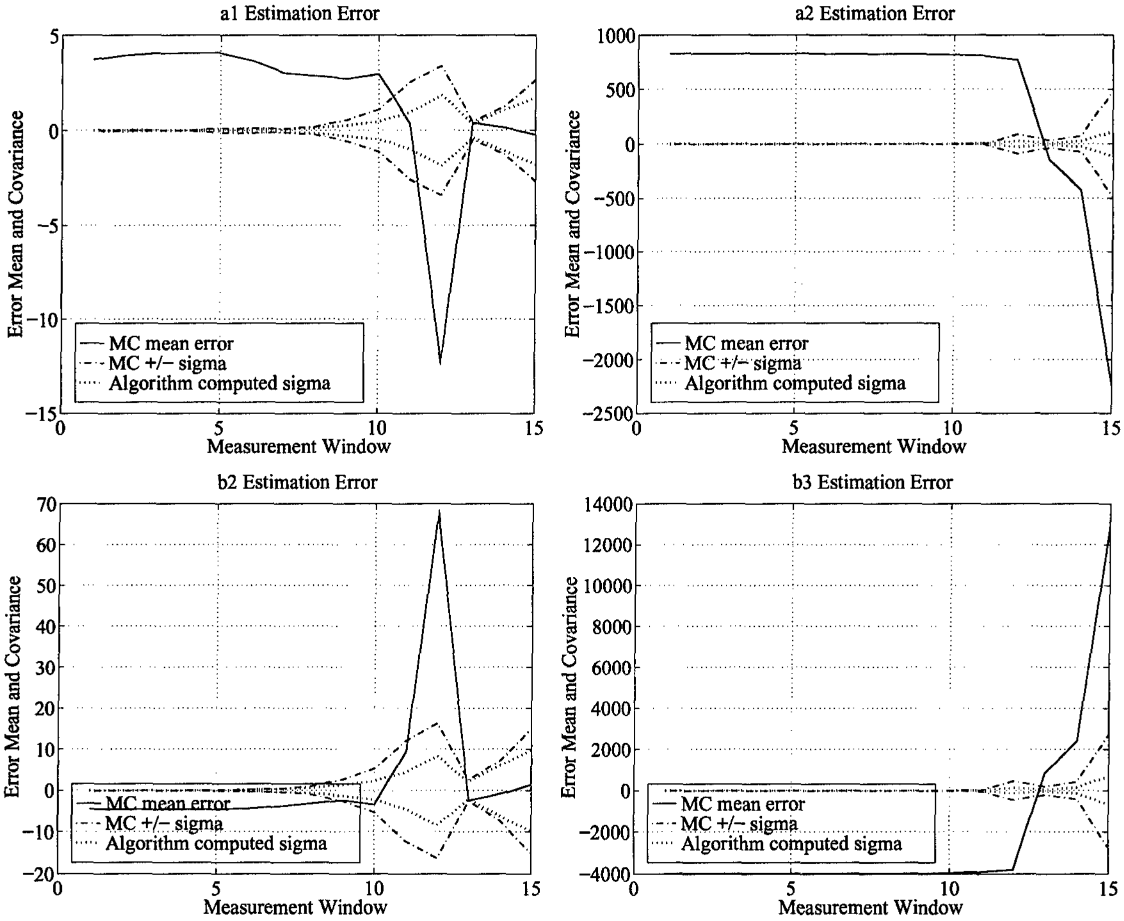


Figure 4.9. Overmodeling Results

4.6 Conclusions

Some degree of *a priori* knowledge about a plant must exist for accurate identification. In the context of this work, this entails knowledge of the plant's bandwidth, and of the bandwidth of the dominant mode.

When one attempts to obtain a second-order model for a second-order plant, the measurements should be in the region of the mode present in the plant or an accurate estimate will not be possible. In the discussed flight control scenario, it seems that the optimum frequency range for proper estimation is immediately prior to the plant's natural frequency. However, this may be specific to this plant and its low frequency zero.

When one attempts to obtain a second-order model of a higher order plant, *i.e.*, when identification in the presence of modeling error is performed, the importance of applying sinusoidal test signals whose frequency is near the frequency of the desired mode becomes greater. Otherwise, not only will the estimation process give you an inaccurate (biased) estimate, but, in addition, it will tell you that it is an accurate estimate. This would lead to placing undue confidence in the estimate and is akin to the notorious filter divergence phenomenon encountered in extended Kalman filtering.

The best parameter match seems to be in window number eight in all higher order cases. This is probably due to matching the second-order Bode plot to the higher order one in that frequency range. If a different range had been used, the results would probably have matched in that range instead. Small variances in the matching range produce small changes in the parameters. Therefore, the small biases in the windows around eight are probably due to matching error and not actual estimation error.

V. Identification of a Discrete-Time Dynamical System

While the identification of continuous-time plants from frequency data is very useful, the primary scenario for the identification of dynamical systems lies in the time domain. In the control systems paradigm, the parameters of the plant are needed to design compensators to elicit desired responses from the plant. To this end, plant input is usually applied in the present day by computer controller actuators. This brings the discrete-time plants models into play. Control is typically applied through a zero-order hold device to the plant.

In this chapter, the discrete-time parameters of a given physical plant are identified using several methods. A comparison is made between the proposed Generalized Minimum Variance (GMV) and established Generalized Least Squares (GLS) identification algorithms. The Instrumental Variable (IV) method is not presented here, because the results are similar to those observed in Chapter III, where the IV estimates were significantly worse than either the GLS or GMV estimates.

Here, the experimental data consists of a finite number of input and output measurements of a discrete-time plant, which is representative of the sampled pitch dynamics of a transport aircraft. Also, sensor noise has corrupted the measurements of the output. The GLS method is described in Ref. [14], and is basically the same as the method proposed in Ref. [32].

Section 5.1 sets up the time domain problem, and explains the application of the GMV and GLS methods. Section 5.2 presents the actual plant used in the experiments. The algorithms are compared using only noise on the output in Section 5.3. The convergence properties are discussed in Section 5.4. Input noise is added to the problem in Section 5.5. Section 5.6 contains the results when the sampling rate is increased by a factor of four.

5.1 Time Domain Identification Setup

Consider the discrete-time z-domain transfer function given by

$$T(z) = \frac{y(z)}{u(z)} = \frac{b_0 z^n + b_1 z^{n-1} + b_2 z^{n-2} + \cdots + b_{n-1} z + b_n}{z^n - a_1 z^{n-1} - a_2 z^{n-2} - \cdots - a_{n-1} z - a_n} = \frac{\sum_{i=0}^{n} b_i z^{n-i}}{\sum_{i=0}^{n} -a_i z^{n-i}} \tag{5.1}$$

where $a_0 = -1$, and the $2n+1$ parameters $a_1 \ldots a_n$ and $b_0 \ldots b_n$ are unknown and may have a zero value, as discussed in Chapter III. This transfer function can be rewritten in time domain form as

$$\underbrace{\left(q^{-0} - q^{-1}a_1 - \cdots - q^{-n}a_n\right)}_{A(q^{-1})} y(kT) = \underbrace{\left(q^{-0}b_0 + q^{-2}b_2 + \cdots + q^{-n}b_n\right)}_{B(q^{-1})} u(kT) \tag{5.2}$$

where T is the sampling period and q^{-n} is an n step time delay. Letting $y_k = y(kT)$ and $u_k = u(kT)$, Eq. (5.2) can be written in recursive form as follows:

$$y_k = a_1 y_{k-1} + a_2 y_{k-2} + \cdots + a_n y_{k-n} + b_0 u_k + b_1 u_{k-1} + b_2 u_{k-2} + \cdots + b_n u_{k-n} \tag{5.3}$$

To identify the parameters of the discrete time system, one need only solve the following system of $2n+1$ equations

$$\begin{bmatrix} y_k \\ y_{k+1} \\ y_{k+2} \\ y_{k+3} \\ \vdots \\ y_{k+N} \end{bmatrix} = \begin{bmatrix} y_{k-1} & y_{k-2} & \cdots & y_{k-n} & u_k & \cdots & u_{k-n} \\ y_k & y_{k-1} & \cdots & y_{k-n+1} & u_{k+1} & \cdots & u_{k-n+1} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ y_{k-1+N} & y_{k-2+N} & \cdots & y_{k-n+N} & u_{k+N} & \cdots & u_{k-n+N} \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_n \\ b_0 \\ \vdots \\ b_n \end{bmatrix} \tag{5.4}$$

However, the values for y_k and u_k cannot be directly measured. Rather, we can measure the noise corrupted Y_k and U_k, where the noise is assumed to be Gaussian:

$$Y_k = y_k + v_k, \qquad v = \mathcal{N}(0, \sigma_v^2)$$
$$U_k = u_k + w_k, \qquad w = \mathcal{N}(0, \sigma_w^2)$$

This changes Eq. (5.3) to

$$\begin{aligned} Y_k - v_k &= a_1(Y_{k-1} - v_{k-1}) + a_2(Y_{k-2} - v_{k-2}) + \cdots + a_n(Y_{k-n} - v_{k-n}) \\ &\quad + b_0(U_k - w_k) + b_1(U_{k-1} - w_{k-1}) + \cdots + b_n(U_{k-n} - w_{k-n}) \end{aligned} \tag{5.5}$$

or

$$Y_k = a_1 Y_{k-1} + a_2 Y_{k-2} + \cdots + a_n Y_{k-n} + b_0 U_k + b_1 U_{k-1} + b_2 U_{k-2} + \cdots + b_n U_{k-n} + \tilde{v}_k$$

where

$$\tilde{v}_k = v_k - a_1 v_{k-1} - a_2 v_{k-2} - \cdots - a_n v_{k-n} - b_0 w_k - b_1 w_{k-1} - b_2 w_{k-2} - \cdots - b_n w_{k-n} \tag{5.6}$$

This expression is now in a form that can be set up in a statistical Linear Regression equation as in Eq. (2.7), where

$$\mathbf{z} = \begin{bmatrix} Y_k \\ Y_{k+1} \\ \vdots \\ Y_{k+N} \end{bmatrix}$$

$$\mathbf{H} = \begin{bmatrix} Y_{k-1} & Y_{k-2} & \cdots & Y_{k-n} & U_k & U_{k-1} & \cdots & U_{k-n} \\ Y_k & Y_{k-1} & \cdots & Y_{k-n+1} & U_{k+1} & U_k & \cdots & U_{k-n+1} \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ Y_{k-1+N} & Y_{k-2+N} & \cdots & Y_{k-n+N} & U_{k+N} & U_{k-1+N} & \cdots & U_{k-n+N} \end{bmatrix}$$

$$\boldsymbol{\theta} = \begin{bmatrix} a_1 & a_2 & \cdots & a_n & b_0 & b_1 & \cdots & b_n \end{bmatrix}^T$$

$$\boldsymbol{\epsilon} = \begin{bmatrix} \tilde{v}_k \\ \tilde{v}_{k+1} \\ \vdots \\ \tilde{v}_{k+N} \end{bmatrix}$$

As before, the **R** matrix is the expected value of $\boldsymbol{\epsilon\epsilon}^T$, and if the noise is assumed to be white, then the **R** matrix will be an $N \times N$ Toeplitz matrix with a non-zero diagonal and n non-zero off diagonal terms above and below the diagonal.

5.1.1 GLS Estimation. The GLS algorithm is derived through an attempt to match the identification problem to a rigorous least squares formulation. The LS estimate would be accurate if the **R** matrix was equal to a scalar multiple of the identity matrix. This would occur if the $\tilde{v}_k$ series was white.

Assuming there is measurement noise only on the output, the system under consideration can be expressed by

$$y(z) = \frac{B(z)}{A(z)}u(z) + e(z)$$

where $e(z)$ is some white noise sequence. This can be rewritten as in Eq. (5.2):

$$A(q^{-1})y_k = B(q^{-1})u_k + A(q^{-1})e_k$$

However, the disturbance $A(q^{-1})e_k$ is no longer white. To "whiten" the system, the following is considered.

$$A(q^{-1})\left(\frac{y_k}{A(q^{-1})}\right) = B(q^{-1})\left(\frac{u_k}{A(q^{-1})}\right) + e_k$$

Here, the input and output are filtered through $1/A(q^{-1})$, the disturbance is once again white, and a LS estimate can be correctly used. The problem with this formulation is that $A(q^{-1})$ is not known. To overcome this, an iterative procedure is once again used. The initial estimate is simply the LS estimate

$$\widehat{\boldsymbol{\theta}}^{(0)}_{GLS} = \widehat{\boldsymbol{\theta}}_{LS}$$

This estimate is used to estimate the necessary $\widehat{A(q^{-1})}$ and the input and output are then filtered as follows.

$$\tilde{y}_k = \widehat{a}_1\tilde{y}_{k-1} + \widehat{a}_2\tilde{y}_{k-2} + \cdots + \widehat{a}_n\tilde{y}_{k-n} + y_k$$

$$\tilde{u}_k = \widehat{a}_1\tilde{u}_{k-1} + \widehat{a}_2\tilde{u}_{k-2} + \cdots + \widehat{a}_n\tilde{u}_{k-n} + u_k$$

Next, $\tilde{y}_k$ and $\tilde{u}_k$ replace the Y and U in the **H** matrix to form the $\tilde{\mathbf{H}}$ matrix, and the next estimate is calculated as follows:

$$\widehat{\boldsymbol{\theta}}^{(1)}_{GLS} = \left(\tilde{\mathbf{H}}^T\tilde{\mathbf{H}}\right)^{-1}\tilde{\mathbf{H}}^T\mathbf{z}$$

This process of filtering and estimating is repeated until some suitable convergence criteria is met, *i.e.*, $\| \hat{\boldsymbol{\theta}}_{GLS}^{(i)} - \hat{\boldsymbol{\theta}}_{GLS}^{(i-1)} \|$ is sufficiently small. It seems that this method also converges for small enough noise values, provided enough measurements are available. As the noise strength is increased, more and more measurements are needed to consistently obtain an estimate. The GLS method also appears to suffer from the problem of converging to incorrect estimates for low enough signal-to-noise ratios.

5.2 Second Order System

For investigative purposes, attention is now restricted to the general second-order dynamical system given by

$$T(s) = \frac{4.8s + 1.44}{s^2 + 0.84s + 1.44} \tag{5.7}$$

which is representative of a transport aircraft's elevator to pitch rate transfer function [2]. For this problem, it is assumed that the control is being passed through a zero-order hold with a 10 Hz sampling rate. Therefore, the control is constant over the sample interval, and the discrete-time (z-domain) transfer function is (see Appendix A)

$$T(z) = \frac{0.4663z - 0.4525}{z^2 - 1.9056z + 0.9194} \tag{5.8}$$

The input to the system is two sinusoids; the minimum number of sinusoids to be persistently exciting for a second-order system with a zero. Thus, the elevator deflection is

$$u(t) = \sin(3t) + \sin(0.5t)$$

In this scenario, it is assumed that there is noise only on the output, and that the input is known. Therefore, according to the GMV paradigm, the equation error noise vector in Eq. (5.6) is given by

$$\tilde{v}_k = v_k - a_1 v_{k-1} - a_2 v_{k-2}$$

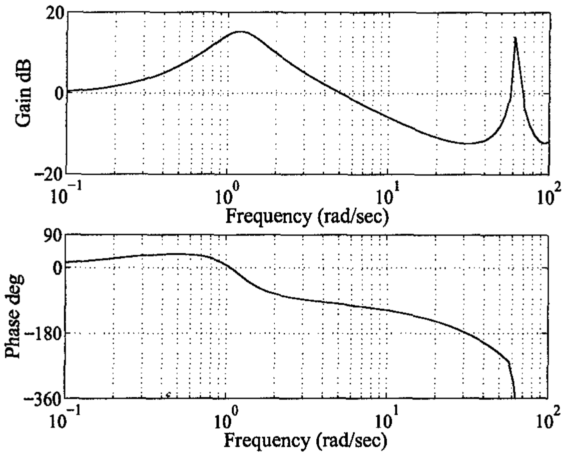


Figure 5.1. Discrete Bode Plot of Given Plant

To obtain the equation error covariance matrix, one calculates

$$E\left\{\tilde{v}_k \tilde{v}_{k-\tau}\right\} = \begin{cases} \sigma_v^2\left(1 + a_1^2 + a_2^2\right), & \tau = 0, \\ \sigma_v^2\left(a_1 a_2 - a_1\right), & \tau = 1, \\ \sigma_v^2\left(-a_2\right), & \tau = 2, \\ 0, & \tau > 2. \end{cases} \tag{5.9}$$

which means that the equation error covariance $\mathbf{R}$ is the pentadiagonal matrix $\mathbf{R} = \sigma_v^2 \mathbf{R}_v$, where

$$\mathbf{R}_v = \begin{bmatrix} 1 + a_1^2 + a_2^2 & -a_1 + a_1 a_2 & -a_2 & 0 & \cdots & 0 \\ -a_1 + a_1 a_2 & 1 + a_1^2 + a_2^2 & -a_1 + a_1 a_2 & -a_2 & \cdots & 0 \\ -a_2 & -a_1 + a_1 a_2 & 1 + a_1^2 + a_2^2 & -a_1 + a_1 a_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 + a_1^2 + a_2^2 \end{bmatrix} \tag{5.10}$$

5.3 Algorithm Comparison

In this section, the GLS and GMV estimates are compared. To do this, Matlab's [22] **randn()** function is initialized with a seed of zero and used to generate noise with a covariance $\sigma_v^2 = (10^p * 0.0001)^2$. This noise is then added to the true output obtained from the system given in Eq. (5.8). Five levels of noise ($p = 0, 1, 2, 3, 4$) are examined in this work, but only three are presented here. The low noise case of $p = 0$ is presented first. The output for this case, corrupted by representative noise, is shown in Fig. 5.2.

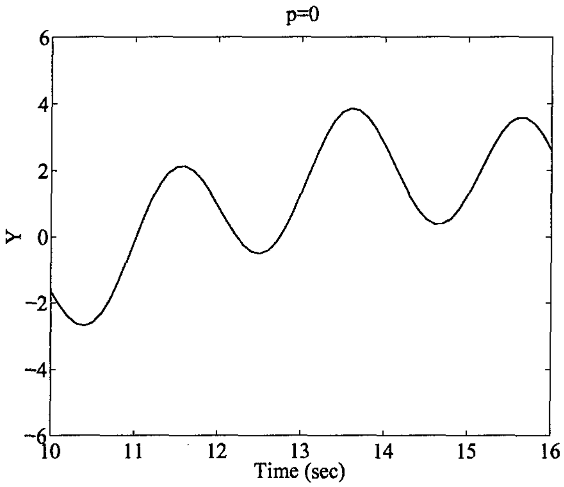


Figure 5.2. Plant Output with Representative Low Noise Level

The $p = 2$ and 3 cases are not presented here because the results are very similar to the first case. Only when the noise level increases to $p = 3$ or 4 do the problems and differences become apparent. The noise levels for these cases are shown in Fig. 5.3.

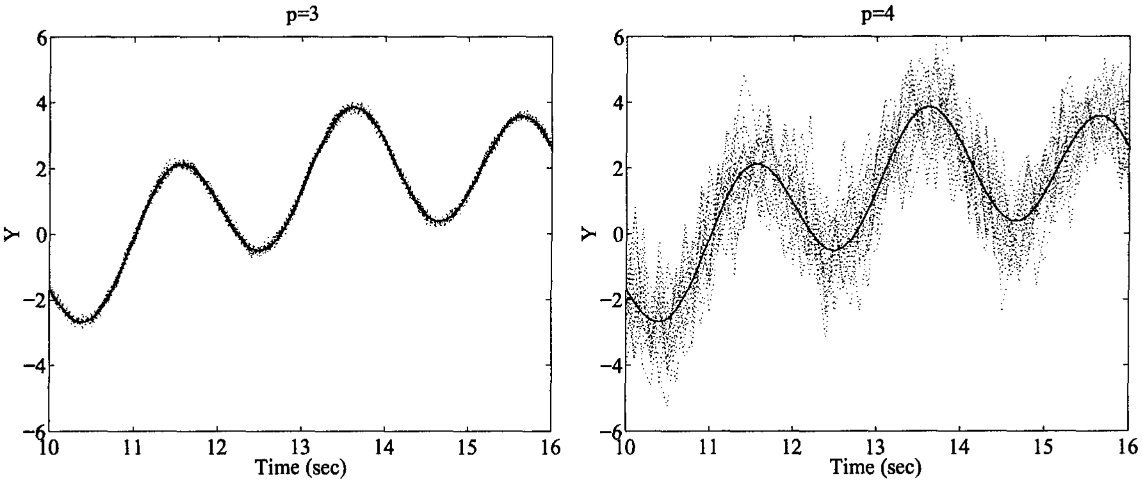


Figure 5.3. Plant Output with Representative High Noise Levels

Figure 5.4 displays the denominator estimates for a 100 run Monte-Carlo (MC) analysis for two different regression lengths - 40 and 70. The estimation results are first normalized by dividing each estimate by the true estimate, and then plotted. Ellipses are plotted representing each estimation method's actual one sigma variation. The ellipses' axes are centered at the average estimate for each method. The algorithm predicted estimation error covariances are not plotted, because they are close to the actual covariances in each case.

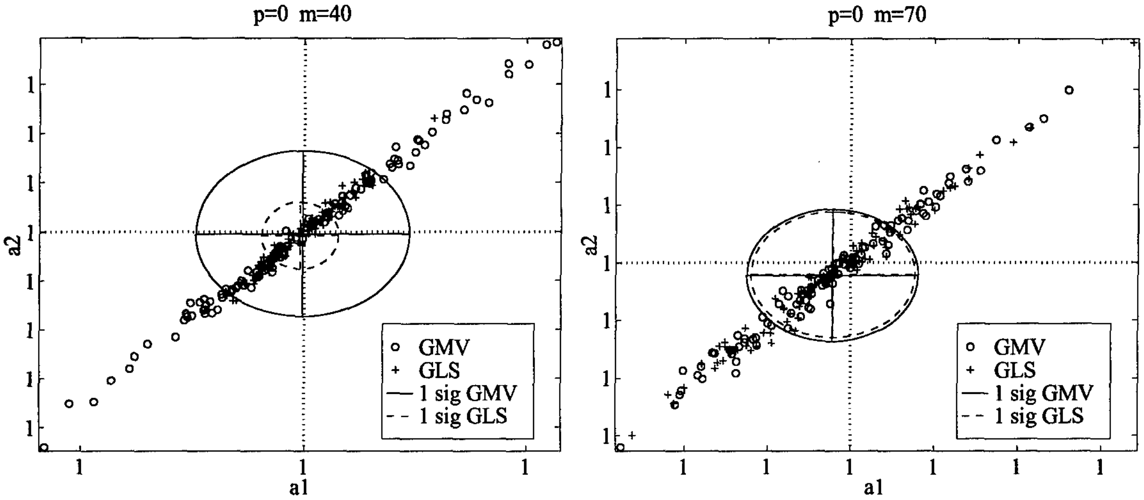


Figure 5.4. Denominator Estimates for 40 (left) and 70 (right) Measurement Linear Regression

As can be seen, the GLS estimates are quite a bit better than the GMV estimates for the case where 40 measurements are taken (approximately 4 seconds of data). However, this does not tell the whole story. When 70 measurements (approximately 7 sec of data) are used, both estimates are better, and the statistical results are virtually the same. To better visualize this behavior, Fig. 5.5 shows the GLS and GMV estimation error and covariance for the a_1 parameter.

As can be seen, the GLS estimate is initially better than the GMV estimate. However, within about 60 measurements, the GMV has caught up with the GLS, and the two

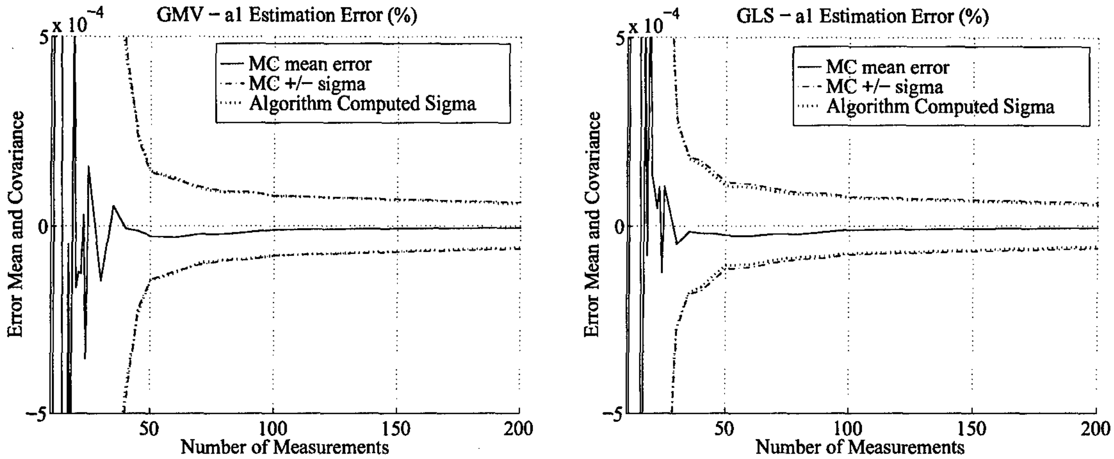


Figure 5.5. GMV and GLS - Incremental a_1 Estimation

are almost identical afterwards. The difference is even smaller when the initial time is removed. Figure 5.6 contains the incremental estimation of the a_1 parameter for the case of no initial time.

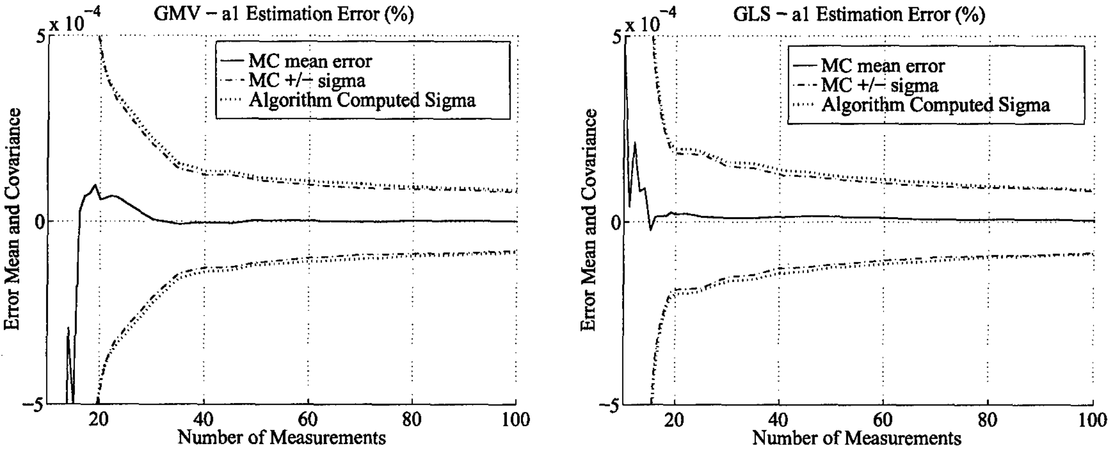


Figure 5.6. GMV and GLS - Incremental a_1 Estimation without Initial Time

From this information, it may seem that the GLS is always at least as good as the GMV, but the cases above are for a very small noise level. When the noise level increases, a problem appears in the GLS estimation. The GLS method requires, in general, more measurements initially to consistently produce a valid estimate. Figure 5.7 contains the incremental a_1 estimation error and covariance for the case of $p = 3$. Once again, it is apparent that the GLS estimate is better than the GMV up until about 60 measurements, but what is not shown is that, while the GMV algorithm produces an estimate for $m < 10$, the GLS algorithm does not produce an estimate for all 100 noise realizations until $m = 19$.

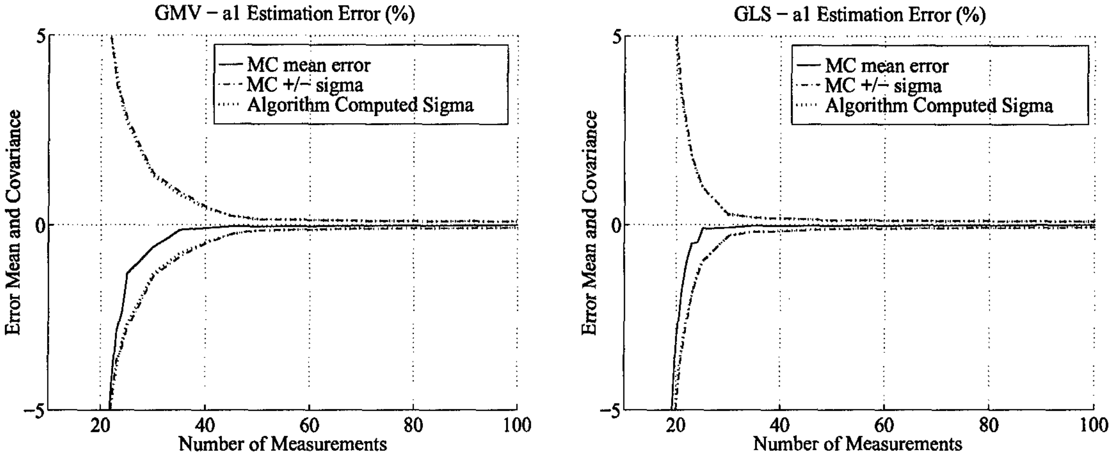


Figure 5.7. Incremental a_1 Estimation for $p = 3$

This initial measurement requirement is more visible when the initial time is removed as well. Figure 5.8 shows these results. The lack of information in the GLS plot below about $m = 25$ is due to the fact that the algorithm does not produce an estimate for this area. Similar things happen for the case of $p = 4$ as well. The GLS estimation requires even more measurements to produce an estimate. The normalized numerical results are given in Appendix D for each of the identification algorithms compared in this work. The values are for $p = 3, 4$ with 400 measurements.

Another difficulty with both methods that begins to play a major role at this noise level is the problem of algorithm convergence. As can be seen in Fig. 5.9, the algorithm predicted covariance is quite far from the actual covariance, and the bias is rather large below 60 measurements for both algorithms. This is due to the fact that the algorithm did

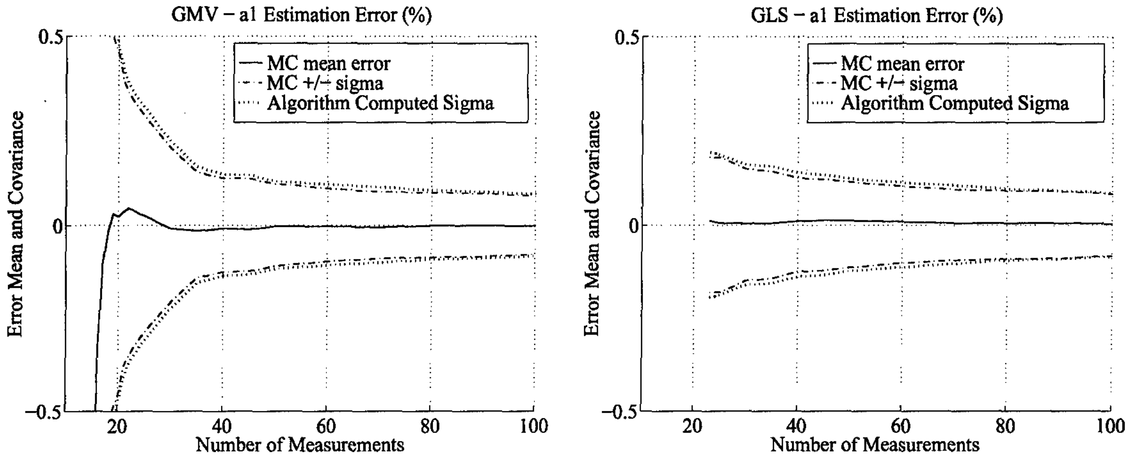


Figure 5.8. Incremental a_1 Estimation for $p = 3$ without Initial Time

not converge to the desired fixed point when initialized with the LS estimate. Rather, it converges to a second fixed point, as is discussed in the next section.

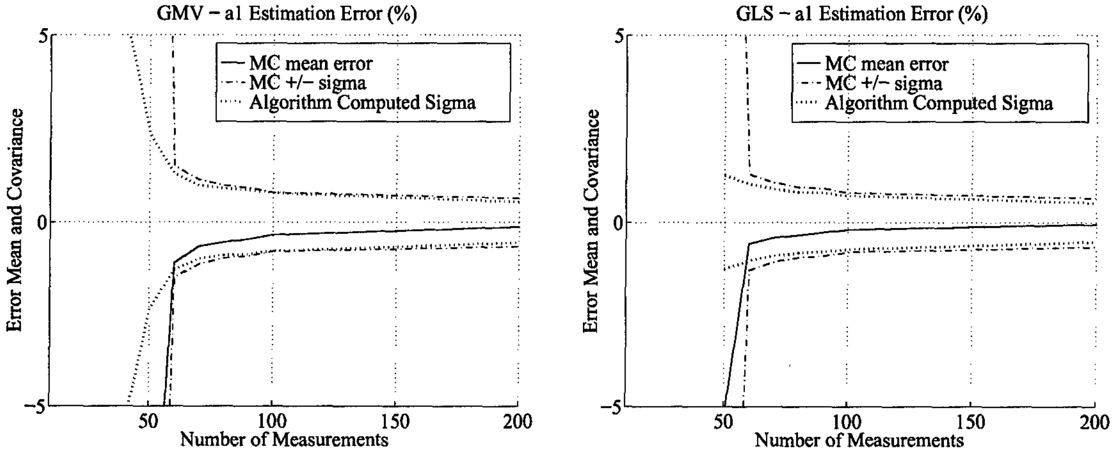


Figure 5.9. Incremental a_1 Estimation for $p = 4$

5.4 Convergence Characteristics

The convergence problem in the time domain case is somewhat worse than that in the frequency domain. In the frequency domain, there was one small area around the origin that, if the algorithm was initialized there, caused the algorithm to converge to an incorrect fixed point. The solution to this problem was just to not initialize in that area, but rather use a large value from each quadrant and choose the one that converged the quickest. However, in the time domain, the picture appears to be much more complicated. Figure 5.10 contains the estimation results for $m = 40$, $p = 4$. As can be seen, both the GMV and GLS algorithms suffer from this problem, and the erroneous fixed point is not necessarily near the origin.

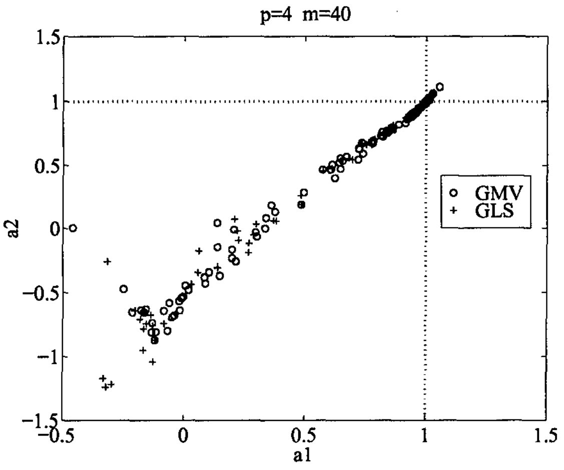


Figure 5.10. Denominator Estimates Showing Convergence Problems

There are at least three different time domain convergence patterns. One is the case where the algorithm converges to the desired estimate no matter where it is initialized. The second case is where the algorithm converges to an incorrect fixed point when initialized at the LS estimate, but will converge to the desired fixed point when initialized close to the true parameter. The third case is where the algorithm will not converge to the desired fixed point no matter where it is initialized. This latter case is truly disconcerting, and it is not known why it happens or how to correct it. The only things that can be said about it are that it only happens for a small number of noise realizations at a very low signal to noise ratio where there are not yet enough measurements. In other words, there is insufficient excitation. The phenomenon disappears once "enough" measurements are added to the linear regression, increasing the level of excitation.

Two examples of case two are shown in Fig. 5.11. They are the 6th and 18th particular noise realizations. The different areas shown in the plots are determined using the "shotgun" approach discussed in Section 3.6.3. The algorithm is initialized at different points, and the thick lines seperate the areas that converge to different points. In realization 6, the small area around the true parameter is the area that yields the desired parameter estimate when the algorithm is initialized there. The rest of the space converges to an incorrect estimate. In realization 18, the upper left and lower right quadrants converge to the desired estimate, and the others do not.

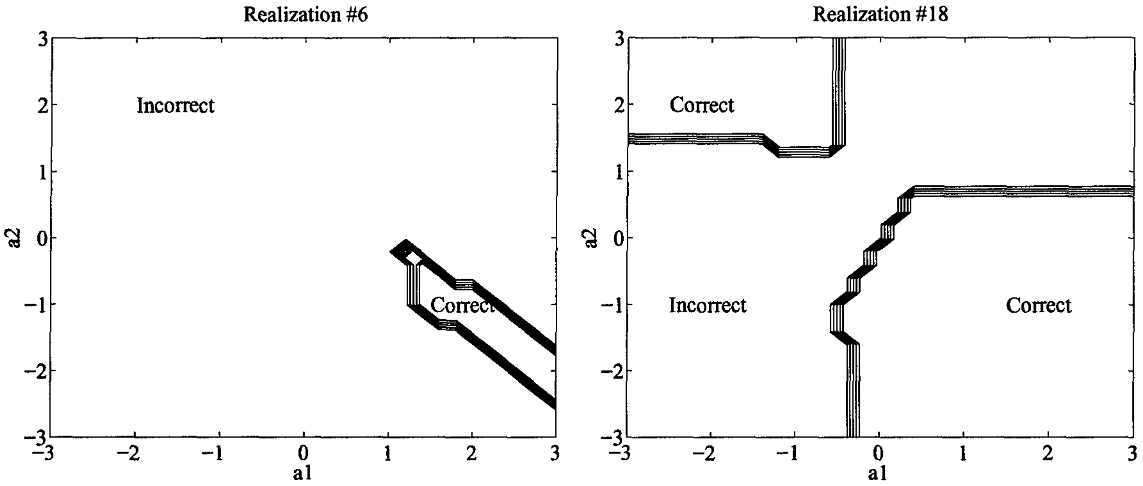


Figure 5.11. Convergence Pattern for Noise Runs 6 and 18

On a brighter note, looking at the output or form of the estimated plants may shed some light on the multiple fixed point quandry. Figure 5.12 contains the true output, the measured output for this realization, and the outputs obtained by running the input through the two different estimated plants. As can be seen, the correct output is much smoother initially than the incorrect one in both cases.

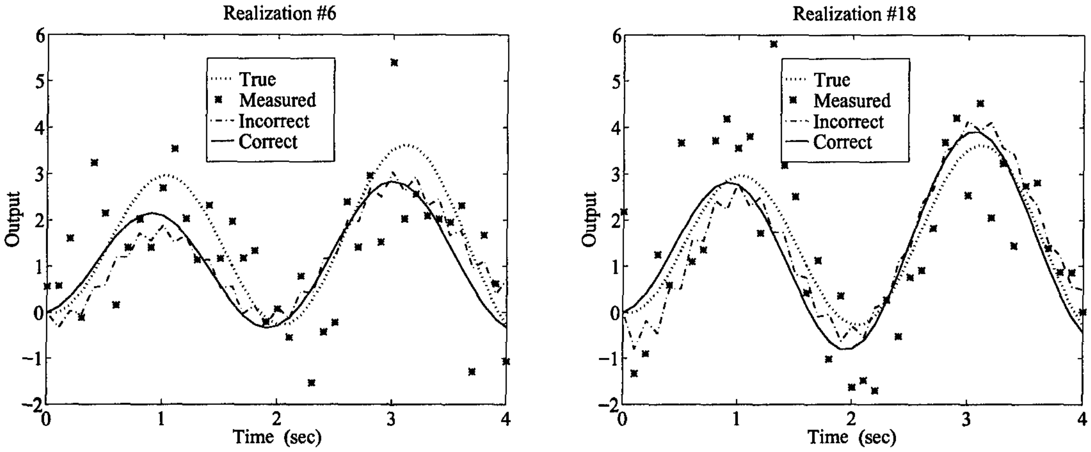


Figure 5.12. Outputs for Noise Realizations 6 and 18

The high measurement noise may fool the identification algorithm into thinking that there is a high-frequency component in the plant. However, upon closer inspection of noise realization number six, it can be seen that the incorrectly identified plant is a highly unlikely candidate. It has one pole on the positive real axis and one on the negative real axis. This is not a possible realization for a physical plant with a zero-order hold, since it would imply one real and one imaginary pole. Additionally, in this case, the pole on the negative real axis is slightly unstable.

$$T(z)_6 = \frac{-0.9477z + 1.6921}{z^2 + 0.2290z - 0.7989} = \frac{-0.9477(z - 1.7854)}{(z + 1.0156)(z - 0.7867)} \tag{5.11}$$

This could also explain, at least in this case, why adding more measurements would eliminate the convergence problems. The more measurements that are added, the less the unstable plant's output would match the measured output.

As for the case of no convergence to the desired parameter, noise realization number 20 is a good example of this. As in the other realizations, there are two fixed points present. Figure 5.13 shows that most of the space converges to an unlikely plant similar to Eq. (5.11), with two poles on the real axis - one positive and one negative.

$$T(z)_{20} = \frac{-1.8521z + 3.2726}{z^2 + 0.3389z - 0.5852} = \frac{-1.8521(z - 1.7670)}{(z + 0.9530)(z - 0.6141)} \tag{5.12}$$

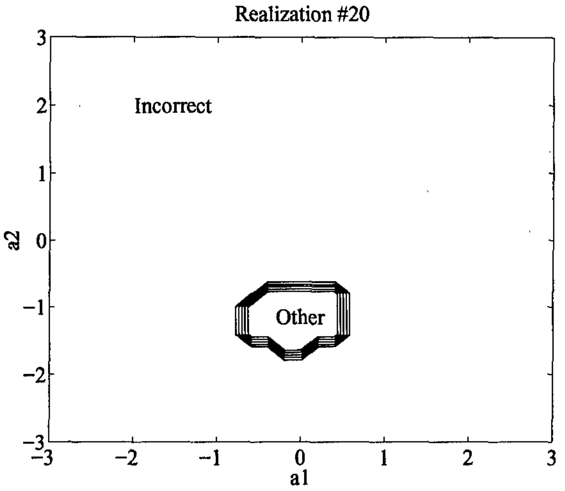


Figure 5.13. Convergence Pattern for Noise Run 20

The small spot, denoted “other”, converges to a completely different plant, but one that is probably as unlikely as the other. This identified plant is given by

$$T(z)_{20b} = \frac{-5.9513z + 8.4410}{z^2 - 0.1807z + 0.9685} = \frac{-5.9513(z - 1.4183)}{(z - 0.0903 \pm j0.98)} \tag{5.13}$$

As can be seen, the imaginary poles are now both plausible for a zero-order hold plant, but the magnitude of the poles is 14.8 rad/sec. If a plant has a mode with a frequency this high, then 0.1 sec is not even close to a sufficiently small sampling period for identification purposes. The output plots in Fig. 5.14 bear this out in that the output for this plant looks very similar qualitatively to the noisy measured output. A high frequency component is readily apparent.

Therefore, some initial knowledge/prior information is necessary if one is trying to identify plants with a very small number of measurements for the noise level present. If the algorithm fails, it does not seem to find physically believable plants for the assumed situation.

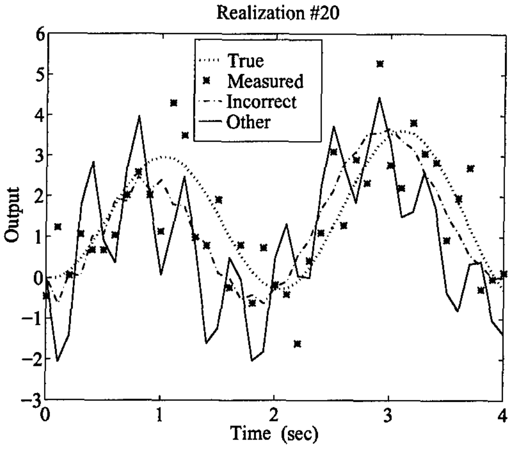


Figure 5.14. Outputs for Noise Realization 20

5.5 Noise on Input

An additional advantage of the GMV method is its adaptability to other linear regression type frameworks, particularly to expanded noise models. If there is noise on the input as well as the output, then the GLS derivation is completely incorrect. The GMV algorithm can, however, be easily adapted to handle input noise simply by modifying $\tilde{v}_k$ to include the new noise. Here,

$$\tilde{v}_k = v_k - a_1 v_{k-1} - a_2 v_{k-2} - b_1 w_{k-1} - b_2 w_{k-2}$$

Now, the needed stochastic modeling entails the calculations

$$E\left\{\tilde{v}_k \tilde{v}_{k-\tau}\right\} = \begin{cases} \sigma_v^2\left(1 + a_1^2 + a_2^2\right) + \sigma_w^2\left(b_1^2 + b_2^2\right), & \tau = 0, \\ \sigma_v^2\left(a_1 a_2 - a_1\right) + \sigma_w^2\left(b_1 b_2\right), & \tau = 1, \\ \sigma_v^2\left(-a_2\right), & \tau = 2, \\ 0, & \tau > 2. \end{cases} \tag{5.14}$$

which means that the equation error covariance matrix $\mathbf{R}$ is now the pentadiagonal matrix $\mathbf{R} = \sigma_v^2 \mathbf{R}_v + \sigma_w^2 \mathbf{R}_w$, where $\mathbf{R}_v$ is given in Eq. (5.10) and the tridiagonal $\mathbf{R}_w$ is as follows:

$$\mathbf{R}_w = \begin{bmatrix} b_1^2 + b_2^2 & b_1 b_2 & 0 & \cdots & 0 \\ b_1 b_2 & b_1^2 + b_2^2 & b_1 b_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & b_1^2 + b_2^2 \end{bmatrix} \tag{5.15}$$

For this comparison, the strength of noise on the input is the same as that on the output. For the case of very small input noise, the GLS-GMV comparison is similar to the case of no input noise. The GLS is better for small m, but the GMV catches up, as is shown in Fig. 5.15. This good performance by GLS is probably due to the filtering effect, *i.e.*, the effects of noise can be reduced by filtering the input and output through low pass filters.

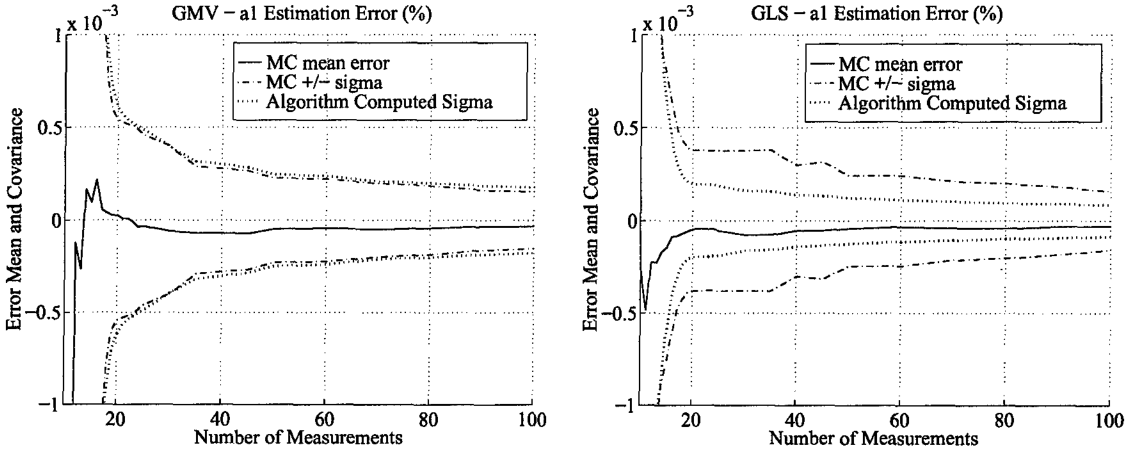


Figure 5.15. Incremental a_1 Estimation for $p = 0$ with Input Noise and No Initial Time

However, when the noise level increases, the input noise seems to have a more pronounced effect, and the GLS estimates become worse than the GMV estimates, as is shown in Figs. 5.16 and 5.17.

Taken to the extreme, the GLS algorithm appears to break down. Not only does it fail to produce an estimate for small m, it also fails for large m. Even when it does produce an estimate, it is not so good except, strangely, for m around 40 to 50. The GMV estimate

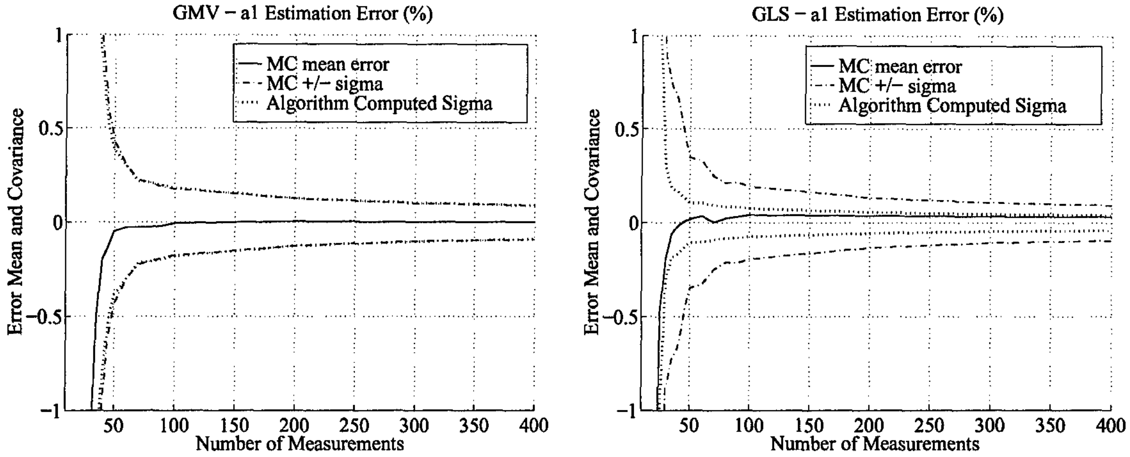


Figure 5.16. Incremental a_1 Estimation for $p = 3$ with Input Noise

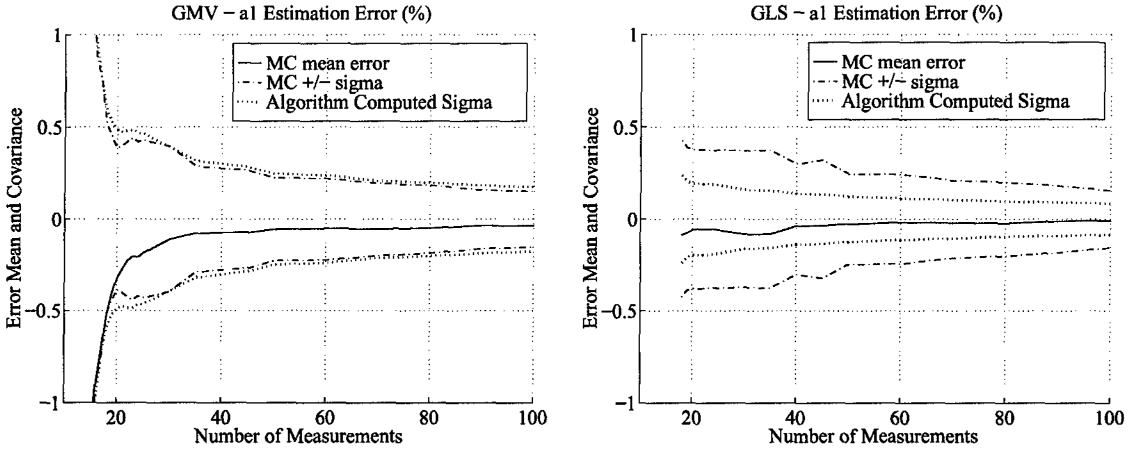


Figure 5.17. Incremental a_1 Estimation for $p = 3$ with Input Noise and No Initial Time

is having trouble with convergence issues for small m, more so with the included initial time case. It takes between 100 and 200 measurements for the initial time case to converge consistently from the LS estimate, while the other case converges consistently for 60 to 70 measurements.

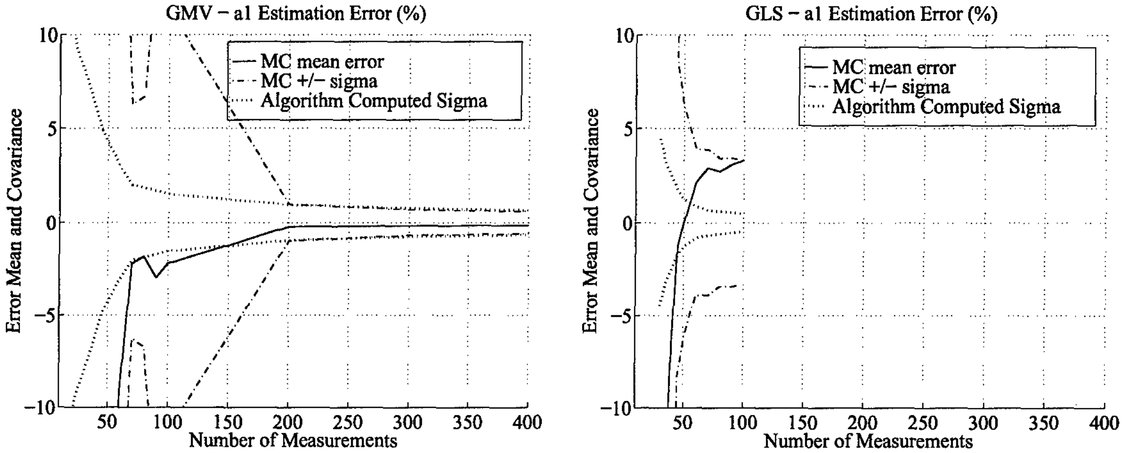


Figure 5.18. Incremental a_1 Estimation for $p = 4$ with Input Noise

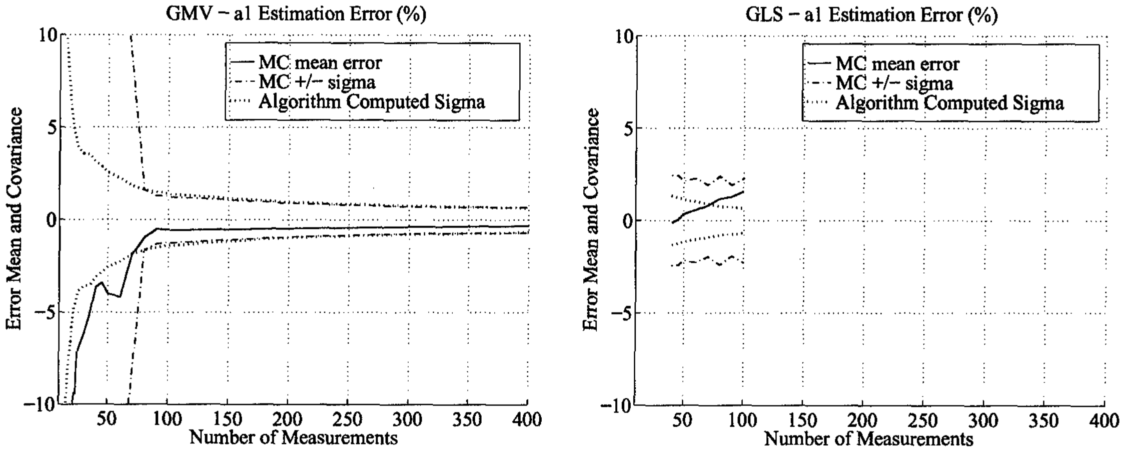


Figure 5.19. Incremental a_1 Estimation for $p = 4$ with Input Noise and No Initial Time

5.6 The Effects of Sampling Rate

In this section, the effects of sampling rate and identification time interval are examined. To do this, the sampling rate for the system is increased by a factor of 4. The new discrete-time transfer function is

$$T(z) = \frac{0.1192z - 0.1183}{z^2 - 1.9783z + 0.9792} \tag{5.16}$$

The incremental identification of one parameter is shown in Fig. 5.20. The left side contains the identification for the original sampling time of 0.1 seconds, and the right hand plot is for the shortened time of 0.025 seconds. As can be seen, the number of measurements in each are such that the physical time interval is the same. It is obvious from the plots that more measurements are needed to obtain the same quality of estimate for the 0.025 rate. Forty measurements at 0.1 produce a relatively good estimate, while the estimate at forty measurements for 0.025 is quite poor. Therefore, increasing the sampling rate will worsen the estimate if the same number of measurements are used.

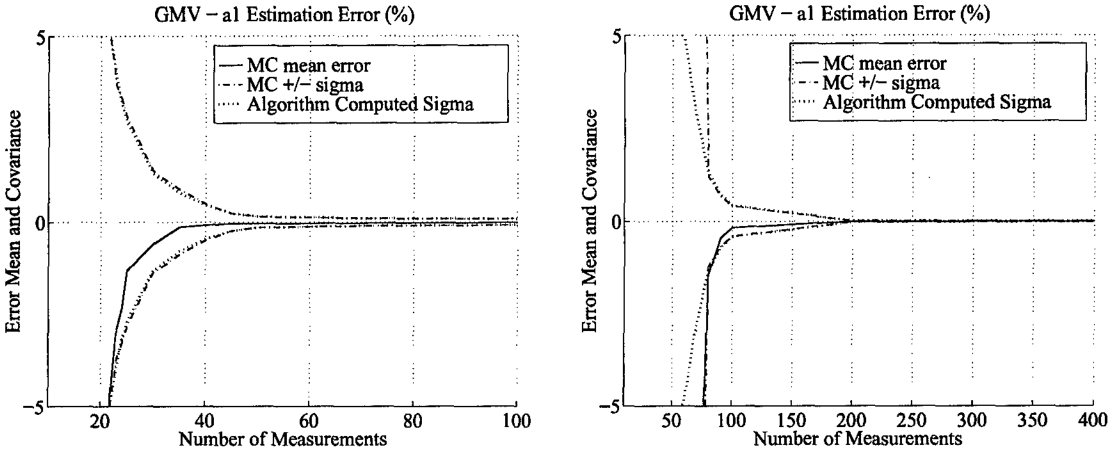


Figure 5.20. Incremental a_1 Estimation for 0.1 (left) and 0.025 (right) Sampling Times

On the other hand, increasing the sampling rate while retaining the same physical identification time interval produces a substantial improvement in the quality of the estimate. Figure 5.21 contains the denominator estimates for 10 seconds of data for both sampling rates. As can be seen, the 0.025 rate, which has 400 measurements, is approximately ten times better than the 0.1 rate, which has only 100 measurements. Conversely, increasing the 0.1 rate estimate to 400 measurements only results in a factor of two improvement. Therefore, if one can afford the computational burden of the increased number of measurements, it is a very good thing to have.

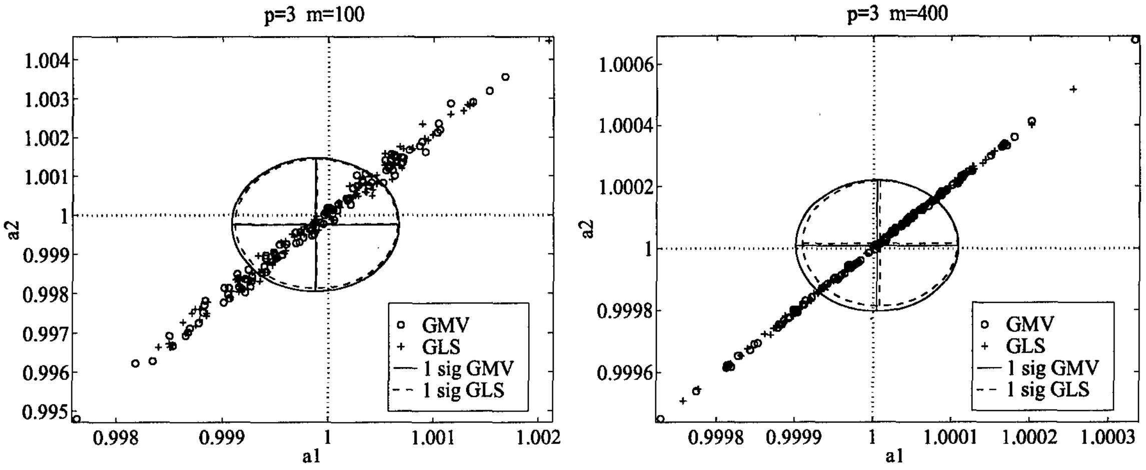


Figure 5.21. Denominator Estimates for 10 Seconds of Data - 0.1 left and 0.025 right

5.7 Conclusions

In this chapter, the performance of the Generalized Minimum Variance and Generalized Least Squares estimation algorithms is compared. Gaussian measurement noise is assumed, as is customary in statistical filtering and system identification work. The novel minimum variance estimate equations are derived and applied to a nonlinear estimation problem. The results are then compared to the established Generalized Least Squares estimate for a second-order system that is representative of an aircraft's pitch dynamics, which is used for inner-loop flight control system design.

While the GLS method requires more measurements initially to obtain an estimate, it appears to produce a better estimate at this point than the GMV. However, the two methods appear to be statistically equivalent as more measurements are added. One distinct advantage of the GMV method is its ability to produce valid estimates at small measurement samples, and in general, under conditions of poorer excitation.

The GMV and GLS estimates outperformed the LS estimate in *all* cases. The LS estimate did, however, provide a useful, albeit sometimes dangerous, starting point for iterating the parameter estimate. At small noise levels, it does not matter where the algorithms are initialized, but as noise levels increase, there is an additional point of convergence that can trap the estimate. In high noise cases (low signal-to-noise ratio), it becomes necessary to examine the final result to see if it is a valid estimate. There seem to be no definitive differences between the convergence rates for incorrect verses correct parameter estimates in the time-domain. Obviously, there must be at least a small amount of prior information about the plant for a valid estimate to be obtained.

As before, the least squares estimate is not effective, even in cases of small measurement noise. The GLS and GMV provide much more accurate estimates, and the GMV seems to provide estimates with much fewer measurements. Additionally, the GMV is much more adaptable to various identification requirements.

VI. Identification of the Physical Parameters of a Continuous-Time Dynamical System

In this chapter, the identification of physical parameters using the sampled input and output of continuous-time plants is discussed. The experiment is based on Case 3 of a system identification competition out of Italy [3], and is related to the work in [5]. These consider the identification of thermal building parameters, which are typically modeled as a second-order thermal circuit with two inputs and one output.

Currently, an important problem of classical system identification is the methods by which algorithms are validated. Too often, the researchers start with a set of input/output data from an unknown plant. They use the data to estimate the parameters of the plant, and then use the input data and the identified plant to simulate the output. If the original output and the simulated output are "close", then the algorithm is considered to be successful. The flaw in this validation "method" becomes apparent under conditions of less than optimal excitation. With poor excitation, it is possible for drastically different plants to produce the same output from the same input. This is a typical manifestation of an inverse problem, which system identification is.

The proposed proper method of algorithm validation is to start with a known plant, produce an input/output data set using this plant, add measurement noise commensurate with the known prevailing signal-to-noise ratio in the actual experiment, identify the parameters, and compare these parameters to the original known plant. If the parameter estimates are good in this "synthetic" experiment, then the algorithm can be applied to the unknown plant as well.

Section 6.1 is a summary of the problem. Section 6.2 explains the method used for identification of the discrete time parameters. Section 6.3 explains the procedure of the experiment. Section 6.4 explains how to use the initial knowledge of the system to improve the identification, and Section 6.5 contains the methods used for obtaining the physical parameter estimates.

6.1 Physical System

The physical system for this experiment is represented by the second-order linear thermal network shown in Fig. 6.1. T_i and T_e are the internal and external temperatures, respectively, Q is the heat flow, R_1, R_2, and R_3 are thermal resistance elements, and G_1 and G_2 are thermal capacitance elements. In the competition, each data set consisted of 30 days of hourly observations of temperature and heat flow. A relatively high level of noise was added to T_e, a low noise level was added to Q, and no noise was added to T_i. The noise on T_e results in a variation of the usual system identification problem, where the input signals are assumed known (noiseless).

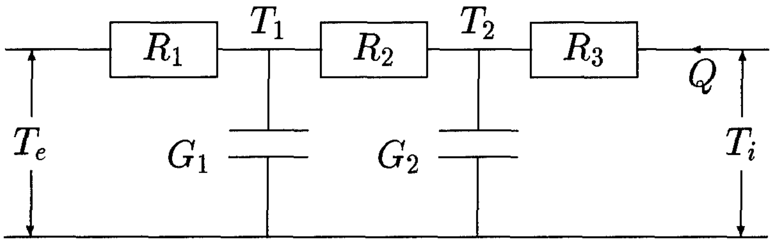


Figure 6.1. Layout of the Physical System

The two states in this system are T_1 and T_2, the two inputs are T_e and T_i, and the output is Q. Therefore, letting $H_i = 1/R_i$, $i = 1,2,3$, the state space equations are

$$\begin{aligned} G_1\frac{dT_1}{dt} &= H_1(T_e - T_1) + H_2(T_2 - T_1) \\ G_2\frac{dT_2}{dt} &= H_2(T_1 - T_2) + H_3(T_i - T_2) \\ Q &= H_3(T_i - T_2) \end{aligned}$$

or in the more familiar form $\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu}$ and $\mathbf{y} = \mathbf{Cx} + \mathbf{Du}$,

$$\begin{bmatrix} \dot{T}_1 \\ \dot{T}_2 \end{bmatrix} = \begin{bmatrix} -\frac{H_1+H_2}{G_1} & \frac{H_2}{G_1} \\ \frac{H_2}{G_2} & -\frac{H_2+H_3}{G_2} \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \end{bmatrix} + \begin{bmatrix} 0 & \frac{H_1}{G_1} \\ \frac{H_3}{G_2} & 0 \end{bmatrix} \begin{bmatrix} T_i \\ T_e \end{bmatrix} \tag{6.1a}$$

$$Q = \begin{bmatrix} 0 & -H_3 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \end{bmatrix} + \begin{bmatrix} H_3 & 0 \end{bmatrix} \begin{bmatrix} T_i \\ T_e \end{bmatrix} \tag{6.1b}$$

This continuous-time system is a Two-Input Single-Output (TISO) system that is not strictly proper. The state equations given above yield the following continuous-time transfer function:

$$Q(s) = \frac{B_0 s^2 + B_1 s + 1}{A_0 s^2 + A_1 s + A_2} T_i(s) - \frac{1}{A_0 s^2 + A_1 s + A_2} T_e(s) \tag{6.2}$$

where

$$B_0 = R_1 R_2 G_1 G_2 \tag{6.3a}$$

$$B_1 = R_1 G_1 + R_2 G_2 + R_1 G_2 \tag{6.3b}$$

$$A_0 = R_1 R_2 R_3 G_1 G_2 \tag{6.3c}$$

$$A_1 = R_1 R_2 G_1 + R_1 R_3 G_1 + R_1 R_3 G_2 + R_2 R_3 G_2 \tag{6.3d}$$

$$A_2 = R_1 + R_2 + R_3 \tag{6.3e}$$

The inputs are T_i and T_e, the output is Q, and the five unknown physical parameters to be identified are R_1, R_2, R_3, G_1, and G_2.

6.2 Time Domain Estimation Method

When a Tustin transformation is performed on a proper or strictly proper second-order continuous system, such as that in Eq. (6.2), the resulting discrete time system is not strictly proper. In other words, the following discrete system is the result.

$$q(z) = \frac{b_{10} z^2 + b_{11} z + b_{12}}{z^2 - a_1 z - a_2} t_i(z) + \frac{b_{20} z^2 + b_{21} z + b_{22}}{z^2 - a_1 z - a_2} t_e(z) \tag{6.4}$$

Here, the discrete-time transfer function's 8 parameters a_1, a_2, b_{10}, b_{11}, b_{12}, b_{20}, b_{21}, and b_{22} are unknown. Although the original continuous-time system had 5 parameters, the Tustin transformation produces 8 in the discrete-time system. This transfer function can be rewritten in recursive form as

$$q_k = a_1 q_{k-1} + a_2 q_{k-2} + b_{10} t_{ik} + b_{11} t_{ik-1} + b_{12} t_{ik-2} + b_{20} t_{ek} + b_{21} t_{ek-1} + b_{22} t_{ek-2} \tag{6.5}$$

If one can obtain the true output q_k, and the true inputs t_{ik} and t_{ek}, then only eight measurements are needed to solve for the unknown parameters. However, the true values cannot be directly measured. Rather, the noise corrupted Q_k, T_{ik}, and T_{ek} are measured, where the noise is assumed temporally uncorrelated (white) and Gaussian:

$$\begin{aligned} Q_k &= q_k + v_k, & v &= \mathcal{N}(0, \sigma_v^2) \\ T_{ik} &= t_{ik} + w_{ik}, & w_i &= \mathcal{N}(0, \sigma_{wi}^2) \\ T_{ek} &= t_{ek} + w_{ek}, & w_e &= \mathcal{N}(0, \sigma_{we}^2) \end{aligned}$$

which, when substituted into Eq. (6.5), yields

$$\begin{aligned} Q_k - v_k = {} & a_1(Q_{k-1} - v_{k-1}) + a_2(Q_{k-2} - v_{k-2}) + b_{10}(T_{ik} - w_{ik}) + b_{11}(T_{ik-1} - w_{ik-1}) \\ & + b_{12}(T_{ik-2} - w_{ik-2}) + b_{20}(T_{ek} - w_{ek}) + b_{21}(T_{ek-1} - w_{ek-1}) + b_{22}(T_{ek-2} - w_{ek-2}) \end{aligned}$$

or

$$Q_k = a_1 Q_{k-1} + a_2 Q_{k-2} + b_{10} T_{ik} + b_{11} T_{ik-1} + b_{12} T_{ik-2} + b_{20} T_{ek} + b_{21} T_{ek-1} + b_{22} T_{ek-2} + \tilde{v}_k \quad (6.6)$$

where

$$\tilde{v}_k = v_k - a_1 v_{k-1} - a_2 v_{k-2} - b_{10} w_{ik} - b_{11} w_{ik-1} - b_{12} w_{ik-2} - b_{20} w_{ek} - b_{21} w_{ek-1} - b_{22} w_{ek-2}$$

This expression is now in a form that can be set up in a statistical linear regression equation given by Eq. (2.7), where

$$\mathbf{z} = \begin{bmatrix} Q_k \\ Q_{k+1} \\ \vdots \\ Q_{k+N} \end{bmatrix}$$

$$\mathbf{H} = \begin{bmatrix} Q_{k-1} & Q_{k-2} & T_{ik} & T_{ik-1} & T_{ik-2} & T_{ek} & T_{ek-1} & T_{ek-2} \\ Q_k & Q_{k-1} & T_{ik+1} & T_{ik} & T_{ik-1} & T_{ek+1} & T_{ek} & T_{ek-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ Q_{k-1+N} & Q_{k-2+N} & T_{ik+N} & T_{ik-1+N} & T_{ik-2+N} & T_{ek+N} & T_{ek-1+N} & T_{ek-2+N} \end{bmatrix}$$

$$\boldsymbol{\theta} = \begin{bmatrix} a_1 & a_2 & b_{10} & b_{11} & b_{12} & b_{20} & b_{21} & b_{22} \end{bmatrix}^T$$

$$\boldsymbol{\epsilon} = \begin{bmatrix} \tilde{v}_k \\ \tilde{v}_{k+1} \\ \vdots \\ \tilde{v}_{k+N} \end{bmatrix}$$

The **R** matrix is the expected value of $\epsilon\epsilon^T$, and if the noise is assumed white, then the **R** matrix is an $N \times N$ Toeplitz matrix with a non-zero diagonal and 2 non-zero off-diagonal terms above and below the diagonal. If v and w_e are assumed uncorrelated and the input measurement T_i is noiseless ($\sigma_{wi} = 0$), then the **R** matrix is given by

$$\mathbf{R} = \sigma_v^2 \begin{bmatrix} 1 + a_1^2 + a_2^2 & -a_1 + a_1a_2 & -a_2 & \cdots & 0 \\ -a_1 + a_1a_2 & 1 + a_1^2 + a_2^2 & -a1 + a_1a_2 & \cdots & 0 \\ -a_2 & -a_1 + a_1a_2 & 1 + a_1^2 + a_2^2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 + a_1^2 + a_2^2 \end{bmatrix}$$

$$+\sigma_{we}^2 \begin{bmatrix} b_{20}^2 + b_{21}^2 + b_{22}^2 & b_{20}b_{21} + b_{11}b_{21} & b_{20}b_{22} & \cdots & 0 \\ b_{20}b_{21} + b_{11}b_{21} & b_{20}^2 + b_{21}^2 + b_{22}^2 & b_{20}b_{21} + b_{11}b_{21} & \cdots & 0 \\ b_{20}b_{22} & b_{20}b_{21} + b_{11}b_{21} & b_{20}^2 + b_{21}^2 + b_{22}^2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & b_{20}^2 + b_{21}^2 + b_{22}^2 \end{bmatrix}$$

6.3 Problem Specifics

The inputs used in this problem are created by taking the first five or six peak frequencies from a Fourier analysis of the competition inputs. Therefore, the inputs used in this experiment are just a sum of cosines and are given by

$$T_\alpha = \sum_{k=1}^{6} M_{\alpha k} \cos\left(\omega_{\alpha k} t + \phi_{\alpha k}\right), \qquad \alpha = i, e$$

where M, ω, and ϕ are given in Table 6.1.

Table 6.1. Values for the T_i and T_e Inputs

k	M_{ik}	ω_{ik}	ϕ_{ik}	M_{ek}	ω_{ek}	ϕ_{ek}
1	25.4132	0	0	11.0476	0	0
2	-0.9391	0.0245	-2.5171	-2.0109	0.0123	-0.9026
3	-0.7854	0.0368	-2.6255	-0.8515	0.0245	-2.3968
4	-0.8098	0.2454	-3.0426	-2.7772	0.2577	-3.0037
5	-2.3719	0.2577	2.9906	-1.6586	0.2700	0.6751
6	-1.2646	0.2700	0.3943	0	0	0

These inputs are given to the system described in Eq. (6.2), where the physical parameters are those of the competition: $R_1 = 1$, $R_2 = 0.1$, $R_3 = 10$, $G_1 = 100$, and $G_2 = 50$. Next, 1000 periods of data are taken, but only the final 402 ($m = 400$) points are used. This is done in an attempt to simulate a building that is in steady state.

Finally, a low noise level is added to the output, and a higher noise level is added to the external temperature input. The signal to noise ratios are based on the competition, and are chosen to be 130 dB and 50 dB, respectively.

6.4 A Priori Information

If one attempts to "brute force" identify all eight of the parameters in Eq. (6.4), then the T_e numerator results are very poor. This is probably because the discrete system is over-determined. There are, in actuality, only five parameters in the system. Keeping this in mind, one can modify the system as follows so only five parameters need be identified.

6.4.1 Modification of the Known Continuous-Time System. Rather than using brute force, it is useful to use the a prior knowledge that one has. To this end, rewrite the transfer functions in Eq. (6.2) as

$$Q(s) = \frac{(B_0 s^2 + B_1 s)\, T_i(s)}{A_0 s^2 + A_1 s + A_2} + \frac{T_i(s) - T_e(s)}{A_0 s^2 + A_1 s + A_2} \tag{6.7}$$

When a Tustin transformation is performed on this system, the following system is the result.

$$q(z) = \frac{b_{10}z^2 + b_{11}z + b_{12}}{z^2 - a_1 z - a_2} t_i(z) + \frac{b_{20}z^2 + b_{21}z + b_{22}}{z^2 - a_1 z - a_2} [ti(z) - t_e(z)] \tag{6.8}$$

It would appear that there are still eight parameters to identify, but when a Tustin transformation is performed on a second-order continuous-time transfer function with a constant numerator, as in the second term of Eq. (6.8), $b_{21} = 2b_{20}$ and $b_{22} = b_{20}$. Also, when a Tustin transformation is performed on a second-order numerator with the form $Ks(s+z)$, as in the first term of Eq. (6.8), $b_{11} + b_{12} + b_{13} = 0$. Therefore, Eq. (6.8) is really

$$q(z) = \frac{(-b_{11} - b_{12})z^2 + b_{11}z + b_{12}}{z^2 - a_1 z - a_2} t_i(z) + \frac{b_{20}z^2 + 2b_{20}z + b_{20}}{z^2 - a_1 z - a_2} [ti(z) - t_e(z)] \tag{6.9}$$

which changes Eq. (6.6) to

$$\begin{aligned} Q_k &= a_1 Q_{k-1} + a_2 Q_{k-2} + b_{11}(T_{ik-1} - T_{ik}) + b_{12}(T_{ik-2} - T_{ik}) \\ &\quad + b_{20}\left((T_{ik} - T_{ek}) + 2(T_{ik-1} - T_{ek-1}) + (T_{ik-2} - T_{ek-2})\right) + \tilde{v}_k \end{aligned} \tag{6.10}$$

where

$$\begin{aligned} \tilde{v}_k &= v_k - a_1 v_{k-1} - a_2 v_{k-2} - (-b_{11} - b_{12} + b_{20}) w_{ik} - (b_{11} + 2b_{20}) w_{ik-1} \\ &\quad - (b_{12} + b_{20}) w_{ik-2} + b_{20} w_{ek} + 2b_{20} w_{ek-1} + b_{20} w_{ek-2} \end{aligned}$$

The number of discrete parameters to be estimated has been decreased to five, and is now equal to the number of unknown physical parameters in the original system. This changes the **H** and **R** matrices to ($\sigma_{wi} = 0$ and $T_{\Delta k} = T_{ik} - T_{ek}$)

$$\mathbf{H} = \begin{bmatrix} Q_{k-1} & Q_{k-2} & (T_{ik-1} - T_{ik}) & (T_{ik-2} - T_{ik}) & (T_{\Delta k} + 2T_{\Delta k-1} + T_{\Delta k-2}) \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

$$\mathbf{R} = \sigma_v^2 \begin{bmatrix} 1 + a_1^2 + a_2^2 & -a_1 + a_1 a_2 & -a_2 & \cdots & 0 \\ -a_1 + a_1 a_2 & 1 + a_1^2 + a_2^2 & -a1 + a_1 a_2 & \cdots & 0 \\ -a_2 & -a_1 + a_1 a_2 & 1 + a_1^2 + a_2^2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 + a_1^2 + a_2^2 \end{bmatrix}$$

$$+\sigma_{we}^2 b_{20}^2 \begin{bmatrix} 6 & 4 & 1 & \cdots & 0 \\ 4 & 6 & 4 & \cdots & 0 \\ 1 & 4 & 6 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 6 \end{bmatrix}$$

6.4.2 Feasibility of the Tustin Transformation. Of concern in this work is the validity of the Tustin transformation. The Tustin transformation assumes ramp type inputs, and substitutes a bilinear approximation for $\exp(\mathbf{A}t)$ in the conversion process (see Appendix A). The inputs to the discrete Tustin system would be the same as the clean sampled inputs to the continuous system. Therefore, the difference would be in the sampled outputs. To try to determine this difference, the discrete system in Eq. (6.9) is simulated with a sampled version of the continuous input that is used with the continuous-time system. The results indicate that there is a discrepancy between the discrete and continuous outputs as shown in Fig. 6.2. As is seen, there is a relatively large initial difference between the two system outputs if the entire data set is used. This agrees with results discussed in Ref. [13]. This difference is probably due to the initial transient in the continuous system, which is

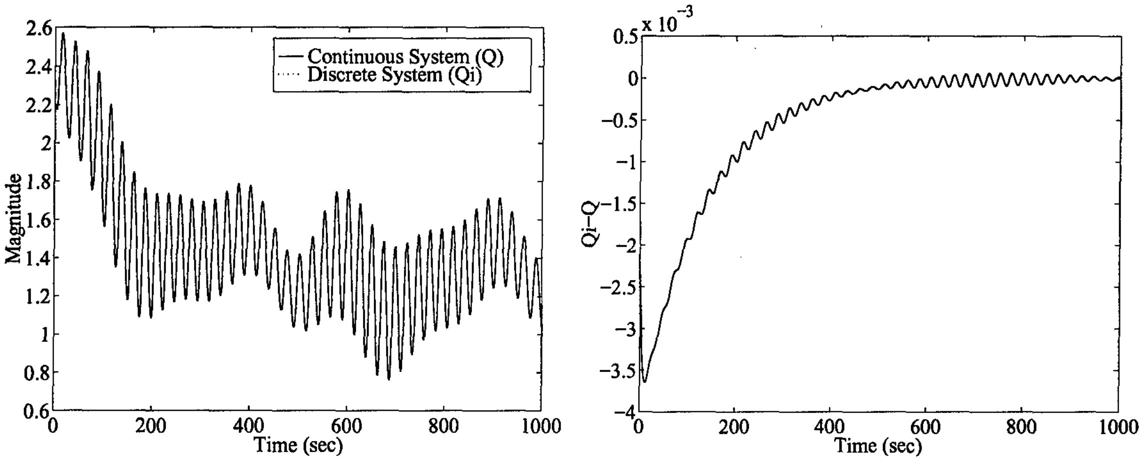


Figure 6.2. Comparison of Continuous and Discrete Outputs

too fast for the Tustin transformation to model. The large initial difference introduces a relatively large error in the estimate, and it takes a very large estimation window to wash it out.

However, if the system measurements are taken after the system is in steady state, the initial differences are smaller. If the corresponding discrete system is simulated using only the information in the final 402 measurements, then the results in Fig. 6.3 are obtained.

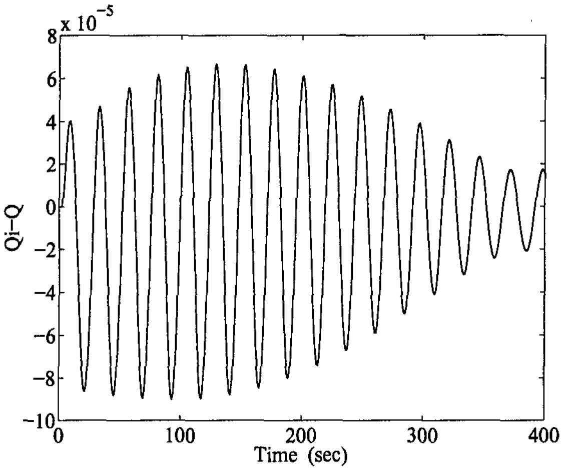


Figure 6.3. Difference with Initial Time Removed

To examine the possible effects of these continuous/discrete differences, the identification experiment is run with a very small measurement noise on both the T_e input and the output, for both the continuous and corresponding discrete systems. The identification results for the continuous system are shown in Fig. 6.4, and the discrete results are in Fig. 6.5. As can be seen, the induced biases are much less than 1% , so the Tustin transformation is considered satisfactory for this problem. If the sampling rate were lower, or if the initial time information was needed, then perhaps a different continuous-discrete transformation would be needed. Other noise models are investigated in this work as well in an attempt to reduce the error.

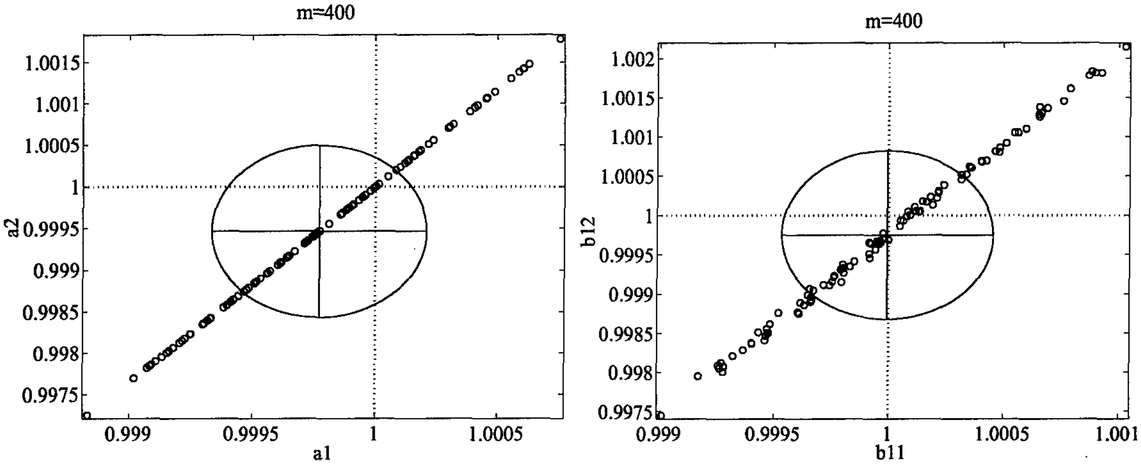


Figure 6.4. Continuous Estimation Results for Very Small Noises

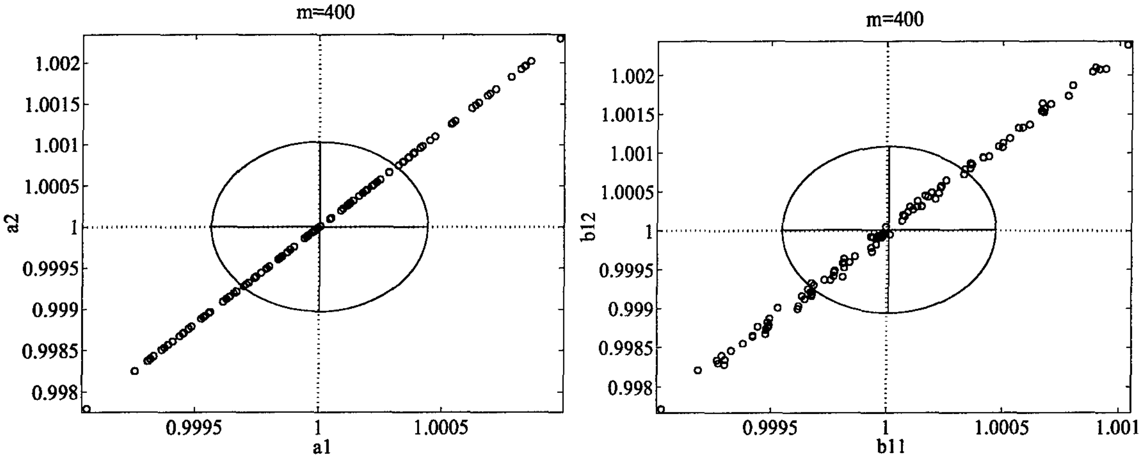


Figure 6.5. Discrete Estimation Results for Very Small Noises

It can be seen in Fig. 6.3 that the difference between the continuous and discrete outputs consists mainly of a high frequency sinusoid and also possibly of a ramp in the beginning. There are at least three possible ways to treat this. One could model the disturbance as a sinusoidal error, a combination sinusoid/ramp, or simply as white noise. Each of the three were added to the system model, but the extra parameters added by the additional modeling seem to negate any gains, and little, if any, improvement is achieved. Therefore, in order to keep the number of estimated parameters to a minimum, a simple white noise model is used.

6.5 Estimation of the Physical System Parameters

This section discusses the procedure used to obtain the required estimates. Once the discrete system parameters and error covariances are estimated, they are converted into the estimates and covariances for G_1, G_2, R_1, R_2, and R_3. To accomplish this, the discrete parameters are first converted to their continuous-time equivalents via an inverse Tustin transformation, and then a set of nonlinear equations is solved for the physical parameters. The covariance estimates are obtained via perturbation methods.

6.5.1 Identification of the Parameters of the Discretized System. As discussed in Section 6.4.2, only the final 402 points are used from the data sets to simulate a building in steady-state. Next, a 100 run Monte Carlo analysis is performed. For each, the least squares estimate is calculated as an initial guess, and the Generalized Minimum Variance estimate is iterated fifty times. Figure 6.6 shows the average estimate, each of the 100 estimates, and the calculated covariance for both of the denominator parameters and two of the numerator parameters. Table 6.2 contains the normalized numerical results for the bias (e), actual error covariance (σ_e), and algorithm predicted covariance (σ_p) for all the parameters.

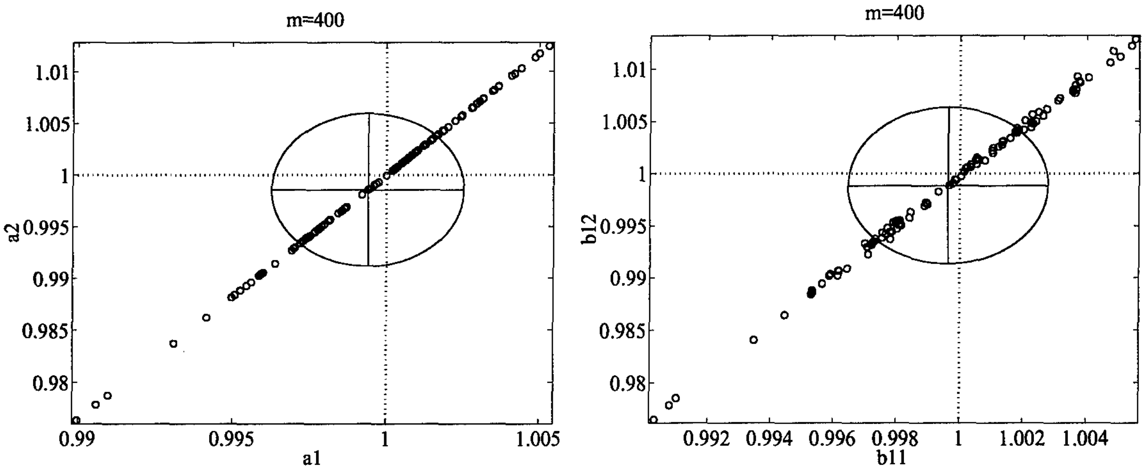


Figure 6.6. Discrete Parameter Estimates and Error Covariances

Table 6.2. Normalized Numerical Results for Discrete Parameters

θ	$e = \widehat{\theta} - \theta$ (%)	σ_e	σ_p
a_1	-0.0605	0.3138	0.2943
a_2	-0.1422	0.7382	0.6921
b_{11}	-0.0378	0.3170	0.2979
b_{12}	-0.1141	0.7497	0.7039
b_{20}	0.3790	1.8682	1.7600

6.5.2 Conversion to Continuous-Time. The first order of business here is to use an inverse Tustin transformation to convert the discrete parameters of Eq. (6.9) into a set of continuous-time parameters as in Eq. (6.2).

The calculation for the error covariance estimate of the continuous-time parameters is done using an approximation of the Jacobian matrix $\tilde{\mathbf{J}}$. To do this, let $\boldsymbol{\theta}_c = [A_0\ A_1\ A_2\ B_0\ B_1]^T$ and $\boldsymbol{\theta}_d = [a_1\ a_2\ b_{11}\ b_{12}\ b_{20}]^T$. Then $\tilde{\mathbf{J}}$ is numerically calculated such that $\Delta\boldsymbol{\theta}_c = \tilde{\mathbf{J}}\,\Delta\boldsymbol{\theta}_d$, where $\Delta\boldsymbol{\theta}_d[i] = \sigma_{\theta d}[i]$. This approximation is exact for linear systems, and it should be close for slightly nonlinear systems. It is assumed that the systems here are slightly nonlinear for the purpose of Jacobian matrix computation. After these calculations are performed, the covariance of the continuous parameters is given by $\mathbf{P}_c = \tilde{\mathbf{J}}\mathbf{P}_d\tilde{\mathbf{J}}^T$. The normalized continuous parameter estimates and covariances are shown in Fig. 6.7, and the normalized numerical results are in Table 6.3.

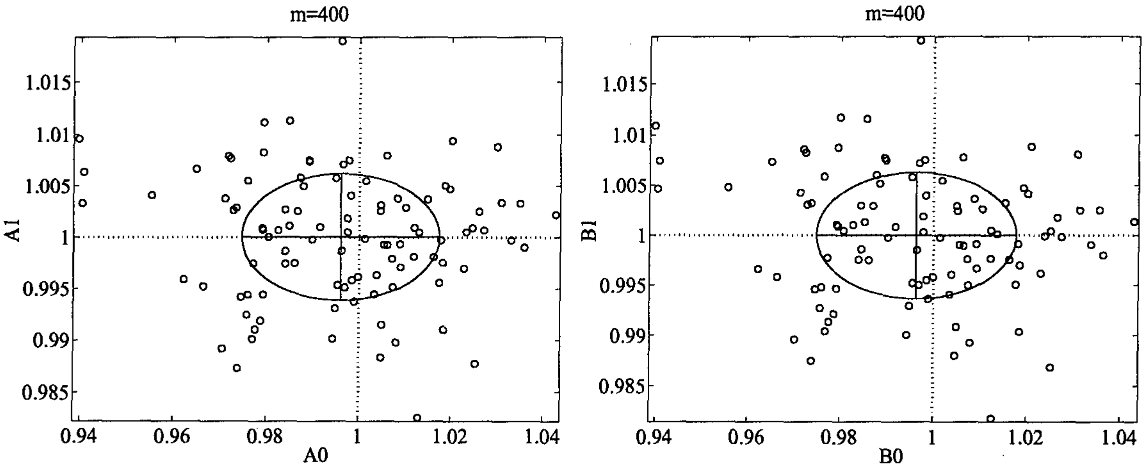


Figure 6.7. Continuous Parameter Estimates and Error Covariances

6.5.3 Conversion to Physical System Parameters. After the estimate and covariance for $\boldsymbol{\theta}_c$ is obtained, the set of nonlinear equations given in Eq. (6.3) are solved for the physical system parameters.

$$R_3 = \frac{A_0}{B_0}$$

Table 6.3. Normalized Numerical Results for Continuous Parameters

θ	$e = \widehat{\theta} - \theta$ (%)	σ_e	σ_p
A_0	-0.3982	2.1402	1.9909
A_1	0.0054	0.6181	0.6456
A_2	0.0044	0.0571	0.0647
B_0	-0.3756	2.1501	2.0015
B_1	-0.0022	0.6326	0.6599

$$C_2 = \frac{B_0^2}{A_1 B_0 - A_0 B_1}$$

$$R_2 = \frac{B_0}{C_2 B_1 - C_2^2(A_2 - R_3)}$$

$$R_1 = A_2 - R_2 - R_3$$

$$C_1 = \frac{B_0}{R_1 R_2 C_2}$$

Once the parameter estimates are obtained, the error covariances are estimated similar to the continuous parameter error covariance estimates, *i.e.*, $\Delta\boldsymbol{\theta} = \tilde{\mathbf{J}}\,\Delta\boldsymbol{\theta}_c$, where $\Delta\boldsymbol{\theta}_c[i] = \sigma_{\theta c}[i]$. The results of these final calculations are shown in Fig. 6.8, and the numerical results are in Table 6.4.

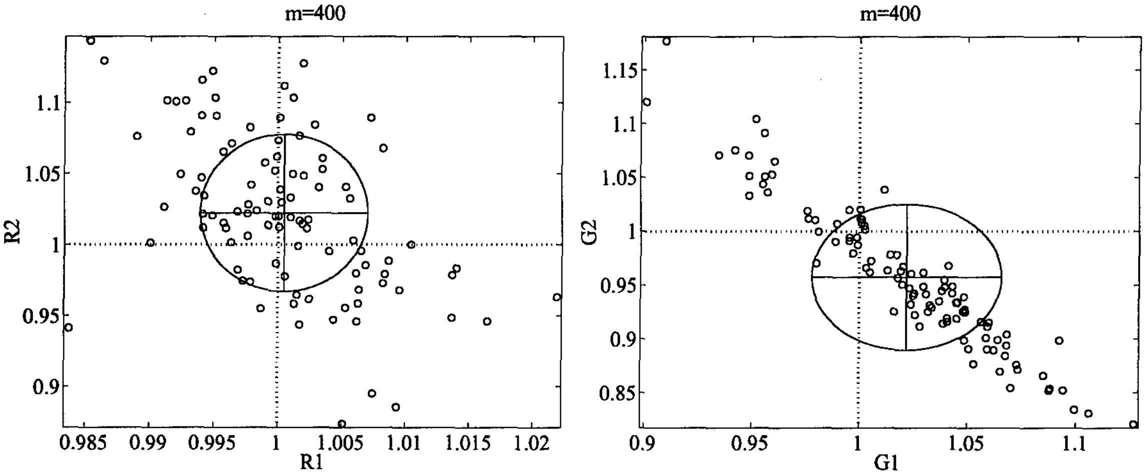


Figure 6.8. System Parameter Estimates and Error Covariances

Table 6.4. Normalized Numerical Results for Physical Parameters

θ	$e = \hat{\theta} - \theta$ (%)	σ_e	σ_p
R_1	0.0479	0.6549	0.9461
R_2	2.2519	5.5391	5.6649
R_3	-0.0224	0.0262	0.0262
G_1	2.1800	4.3873	4.1068
G_2	-4.2527	6.7806	5.2851

6.6 Conclusions

This chapter investigates a method of identifying physical parameters using sampled measurements of continuous-time inputs and outputs. The flexibility of the GMV algorithm is apparent in this problem, adapting easily to measurement noise on one input and on the output. Also, the system has been rearranged such that only five parameters need be identified. The well known Tustin transformation is used to convert the estimated discrete-time parameters to continuous-time parameters. The error covariances are estimated numerically, and the results are close to the actual covariances. The physical parameters and error covariances are calculated in a similar manner.

An important point of this chapter is the number of parameters to be identified. If one tries to identify an over-determined system, the results will suffer, if they can be obtained at all.

VII. Conclusions.

This research identifies and focuses on the solution of the problems that measurement noise introduces to the identification of dynamical systems. Measurement noise is a fact of life in experimental and applied science, and should be specifically addressed. The Generalized Minimum Variance (GMV) identification procedure developed in this work is one possible solution. An advantage of this method is the fact that it is based on linear regression and linear prediction. This makes it easy to understand and adapt to other estimation/system identification problems, where it produces equally good results. Further applications of the GMV algorithm can be found in Ref. [15].

This work introduces the concept that the parameter estimate provided by the GMV algorithm is a fixed point of a nonlinear mapping derived from the theory of Minimum Variance identification. Existence of a fixed point is proven, and the particular algorithm used corresponds to the iteration suggested by a contraction mapping. This eliminates the need for global optimization, and the problems of multiple local minima are greatly reduced. In fact, only one other possible fixed point solution was observed in the course of this work at low signal-to-noise ratios. However, this benign situation may not persist for higher order systems. Moreover, when the alternative solution is found, it is not a plausible one for the given situation, and possible means of eliminating it are suggested.

In Chapter III, the GMV algorithm is applied to the identification of continuous-time transfer functions using frequency domain measurements. Three different noise models are examined, and the results are compared to those from the classical Least Squares, Generalized Least Squares (GLS), and Instrumental Variable methods. Only the GLS algorithm produces similar quality parameter estimates. Moreover, the way in which the GLS method is cost minimization-based perhaps clouds the treatment of measurement noise, and the results suffer. When the noise is properly treated in the GMV method, the results are better. Convergence rates for the two algorithms are similar, with a small number of iterations needed at small noise levels. However, large numbers of iterations are needed as the noise level becomes high.

Prior knowledge of the system order is needed to use the GMV method, and some knowledge of the system bandwidth is needed to obtain the parameter estimates accurately. As is seen in Chapter IV, if ones takes measurements outside the bandwidth of interest, totally inaccurate estimates are obtained.

In Chapter V, the GMV algorithm is applied to the identification of a discrete-time plant. This scenario is applicable to control system design, since the input to a computer controlled plant is typically zero-order hold in nature, and thus one obtains a direct and correct transformation from the continuous-time model to the discrete-time model. The versatility of the GMV identification method is apparent in this situation as it easily adapts to accommodate the addition of measurement noise on the input.

Also readily visible in this chapter is the importance of correct model validation. In general, simply because the estimated and measured outputs are close does not mean that the system has been correctly identified. In the GMV estimation method, an iteration is performed to find a fixed point. As the noise increases, another fixed point appears. When the output of this second estimated plant is compared to the correctly estimated plant, little quantitative difference is apparent. Thus, the output of two distinctly different plants can have similar outputs for a given input. Fortunately, at least in this scenario, the incorrectly identified plant is not a plausible solution.

Finally, Chapter VI applies the GMV algorithm to the identification of a continuous-time dynamical system, and to the estimation of its physical parameters. Here, noise is present on one of the inputs and the output of the second-order, two input-one output plant. The Tustin transformation is determined to be a suitable one in this case, because the deterministic errors introduced are less than one percent. Note, one should not simply apply any identification method blindly. If only actual measurements are available, with no idea of the actual parameters, one should first perform a synthetic experiment and gauge the results. If problems arise in such an experiment, they can then be addressed before one attempts to identify the actual unknown parameters using real data.

An additional important aspect of system identification is also addressed in this chapter: the concept of keeping the number of identified parameters to a minimum. This is where prior knowledge of the system is useful. In the problem addressed in Chapter V, there were actually only five parameters describing the system. If the problem is addressed in the time domain with no prior knowledge, then the number of needed parameters is eight. Using knowledge of the system however, one can reduce the number of needed parameters in the time domain to five - the minimum number. Failing to do so severely hampers the identification process.

In conclusion, the attributes of proper (stochastic) modeling cannot be overemphasized. The performance of the GMV system identification algorithm is shown to be equal to or superior than the best available linear regression based system identification methods. It produces good parameter estimates for small numbers of measurements, along with an accurate prediction of the estimation error, when convergence is not a problem. Convergence of the GMV algorithm occurs for relatively low signal-to-noise ratios. Hence, the GMV algorithm is completely autonomous for reasonable noise levels. Also, the structure of the **R** matrix may allow the design of recursive algorithms if so desired. Finally, being based on proper stochastic modeling, the GMV system identification method developed in this dissertation is shown to be readily adaptable and equally applicable to a whole spectrum of identification and estimation problems.

Appendix A. Methods for Discretizing Continuous Systems

While physical systems are almost always continuous-time systems, they are typically converted into discrete-time models for many purposes, including digital control and system identification. There are many variations in discretization methods, and most are based on differing assumptions about the form of the input.

A.1 Continuous-Time Model

All linear (or linearized) physical systems can be represented by the familiar form

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \tag{A.1a}$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \tag{A.1b}$$

where $\mathbf{x}$ is a vector of the states in the system, $\mathbf{u}$ is a vector of the inputs to the system, and $\mathbf{y}$ is a vector containing the outputs of the system. The goal of discretizing the system is to obtain a model of the form.

$$\mathbf{x}_k = \mathbf{A}_d\mathbf{x}_{k-1} + \mathbf{B}_d\mathbf{u}_{k-1} \tag{A.2a}$$

$$\mathbf{y}_k = \mathbf{C}_d\mathbf{x}_k + \mathbf{D}_d\mathbf{u}_k \tag{A.2b}$$

where $\mathbf{x}_k = \mathbf{x}(kT)$, $\mathbf{x}_{k-1} = \mathbf{x}((k-1)T)$, and likewise for $\mathbf{u}_k$ and $\mathbf{y}_k$. Comparing Eq. (A.2b) to Eq. (A.1b), it can be seen that

$$\mathbf{C}_d = \mathbf{C} \qquad \text{and} \qquad \mathbf{D}_d = \mathbf{D} \tag{A.3}$$

To solve for the time history of the output, given the time history of the input $\mathbf{u}(t)$ and the system states at the initial time $\mathbf{x}_0 = \mathbf{x}(t_0)$, the following is used [7]:

$$\mathbf{x}(t) = \mathrm{e}^{\mathbf{A}(t-t_0)}\mathbf{x}_0 + \int_{t_0}^{t} \mathrm{e}^{\mathbf{A}(t-\tau)}\mathbf{B}\mathbf{u}(\tau)d\tau \tag{A.4}$$

This equation is the primary building block for discretizing the system. Setting the initial time equal to $(k-1)T$, where T is the discretization step, yields

$$\mathbf{x}_k = \mathrm{e}^{\mathbf{A}T}\mathbf{x}_{k-1} + \int_{(k-1)T}^{kT} \mathrm{e}^{\mathbf{A}(kT-\tau)}\mathbf{B}\mathbf{u}(\tau)d\tau \tag{A.5}$$

In order to perform the integration shown in Eq. (A.5), total knowledge of $\mathbf{u}(t)$ for $(k-1)T < t < kT$ is required. Given a sampled input, one has knowledge of $\mathbf{u}(t)$ only at the endpoints, so some assumption must be made about the form of $\mathbf{u}(t)$ in between.

To begin, let $t = kT - \tau$, so $dt = -d\tau$, and Eq. (A.5) becomes

$$\mathbf{x}_k = \mathrm{e}^{\mathbf{A}T}\mathbf{x}_{k-1} + \int_0^T \mathrm{e}^{\mathbf{A}t}\mathbf{B}\mathbf{u}(kT - t)dt \tag{A.6}$$

A.2 Assumption of Constant Input Between Samples

In this scenario, it is assumed that the input is constant and equal to that at the initial time, *i.e.*,

$$\mathbf{u}(t) = \mathbf{u}\left((k-1)T\right) = u_{k-1}, \qquad (k-1)T \leq t < kT$$

Now, since the input is a constant, it can be brought outside the integration, so Eq. (A.6) becomes

$$\mathbf{x}_k = \mathrm{e}^{\mathbf{A}T}\mathbf{x}_{k-1} + \left[\int_0^T \mathrm{e}^{\mathbf{A}t}dt\right]\mathbf{B}\mathbf{u}_{k-1} \tag{A.7}$$

Integrating the term in square brackets yields

$$\int_0^T \mathrm{e}^{\mathbf{A}t}dt = \mathbf{A}^{-1}\mathrm{e}^{\mathbf{A}t}\Big]_0^T = \mathbf{A}^{-1}\left(\mathrm{e}^{\mathbf{A}T} - \mathbf{I}\right) \tag{A.8}$$

so provided that $\mathbf{A}$ is invertible,

$$\mathbf{x}_k = \mathrm{e}^{\mathbf{A}T}\mathbf{x}_{k-1} + \left[\mathbf{A}^{-1}\left(\mathrm{e}^{\mathbf{A}T} - \mathbf{I}\right)\mathbf{B}\right]\mathbf{u}_{k-1} \tag{A.9}$$

Upon inspection, it is apparent that Eq. (A.9) has the same form as Eq. (A.2a), where

$$\mathbf{A}_d = \mathrm{e}^{\mathbf{A}T} \quad \text{and} \quad \mathbf{B}_d = \mathbf{A}^{-1}\left(\mathrm{e}^{\mathbf{A}T} - \mathbf{I}\right)\mathbf{B} \tag{A.10}$$

This can be converted to a discretized transfer function via

$$\frac{y(z)}{u(z)} = \mathbf{C}_d\left(z\mathbf{I} - \mathbf{A}_d\right)^{-1}\mathbf{B}_d \tag{A.11}$$

A.2.1 Example System from Chapter V. The continuous-time transfer function for this system is

$$T(s) = \frac{4.8s + 1.44}{s^2 + 0.84s + 1.44}$$

This can be converted to the control canonical state space form [28]

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 1 \\ -1.44 & -0.84 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) \tag{A.12a}$$

$$y(t) = \begin{bmatrix} 1.44 & 4.8 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \end{bmatrix} u(t) \tag{A.12b}$$

Using Eqs. (A.3) and (A.10), setting $T = 0.1$, the discretized matrices are

$$\mathbf{A}_d = \begin{bmatrix} 0.9930 & 0.0957 \\ -0.1378 & 0.9126 \end{bmatrix} \qquad \mathbf{B}_d = \begin{bmatrix} 0.0049 \\ 0.0957 \end{bmatrix}$$

$$\mathbf{C}_d = \begin{bmatrix} 1.44 & 4.8 \end{bmatrix} \qquad \mathbf{D}_d = \begin{bmatrix} 0 \end{bmatrix}$$

Therefore, using Eq. (A.11), the resulting discretized transfer function is

$$T(z) = \frac{0.4663z - 0.4525}{z^2 - 1.9056z + 0.9194} \tag{A.13}$$

A.3 Assumption of Linear Form Between Input Samples

In this scenario, it is assumed that the input is linear between the two samples. Therefore, the input is given by

$$\begin{aligned} \mathbf{u}(t) &= \left[\frac{t-(k-1)T}{T}\right] \mathbf{u}_k + \left[1 - \frac{t-(k-1)T}{T}\right] \mathbf{u}_{k-1} \\ &= \mathbf{u}_{k-1} + \frac{t-(k-1)T}{T}(\mathbf{u}_k - \mathbf{u}_{k-1}) \end{aligned}$$

so

$$\mathbf{u}(kT - t) = \mathbf{u}_k + \frac{\mathbf{u}_{k-1} - \mathbf{u}_k}{T} t \tag{A.14}$$

Substituting Eq. (A.14) into Eq. (A.6) and simplifying yields

$$\mathbf{x}_k = e^{\mathbf{A}T}\mathbf{x}_{k-1} + \left[\int_0^T e^{\mathbf{A}t} dt\right] \mathbf{B}\mathbf{u}_k + \frac{1}{T}\left[\int_0^T te^{\mathbf{A}t} dt\right] \mathbf{B}(\mathbf{u}_{k-1} - \mathbf{u}_k) \tag{A.15}$$

Integrating the second bracketed term by parts yields

$$\int_0^T t\mathrm{e}^{\mathbf{A}t}dt = \mathbf{A}^{-1}t\mathrm{e}^{\mathbf{A}t}\Big]_0^T - \int_0^T \mathbf{A}^{-1}\mathrm{e}^{\mathbf{A}t}dt = \mathbf{A}^{-1}T\mathrm{e}^{\mathbf{A}T} - \mathbf{A}^{-2}\left(\mathrm{e}^{\mathbf{A}T} - \mathbf{I}\right) \tag{A.16}$$

Now, substituting Eqs. (A.8) and (A.16) into Eq. (A.15),

$$\mathbf{x}_k = \mathbf{A}_d\mathbf{x}_{k-1} + \mathbf{B}_{d0}\mathbf{u}_k + \mathbf{B}_{d1}\mathbf{u}_{k-1}$$

where

$$\mathbf{A}_d = \mathrm{e}^{\mathbf{A}T} \tag{A.17a}$$

$$\mathbf{B}_{d0} = \left[\left(\mathbf{A}^2T\right)^{-1}\left(\mathrm{e}^{\mathbf{A}T} - \mathbf{I}\right) - \mathbf{A}^{-1}\right]\mathbf{B} \tag{A.17b}$$

$$\mathbf{B}_{d1} = \left[\mathbf{A}^{-1}\mathrm{e}^{\mathbf{A}T} - \left(\mathbf{A}^2T\right)^{-1}\left(\mathrm{e}^{\mathbf{A}T} - \mathbf{I}\right)\right]\mathbf{B} \tag{A.17c}$$

As one can see, this system does not have the form of Eq. (A.2). Rather, it has the form

$$\mathbf{x}_k = \mathbf{A}_d\mathbf{x}_{k-1} + \mathbf{B}_{d0}\mathbf{u}_k + \mathbf{B}_{d1}\mathbf{u}_{k-1} \tag{A.18a}$$

$$\mathbf{y}_k = \mathbf{C}_d\mathbf{x}_{k-1} + \mathbf{D}_d\mathbf{u}_{k-1} \tag{A.18b}$$

This can be rewritten in z-domain form as

$$\begin{aligned} z\mathbf{x}(z) &= \mathbf{A}_d\mathbf{x}(z) + z\mathbf{B}_{d0}\mathbf{u}(z) + \mathbf{B}_{d1}\mathbf{u}(z) \\ \mathbf{y}(z) &= \mathbf{C}_d\mathbf{x}(z) + \mathbf{D}_d\mathbf{u}(z) \end{aligned}$$

so

$$\begin{aligned} \mathbf{x}(z) &= \left(z\mathbf{I} - \mathbf{A}_d\right)^{-1}\left(z\mathbf{B}_{d0} + \mathbf{B}_{d1}\right)\mathbf{u}(z) \\ \mathbf{y}(z) &= \mathbf{C}_d\left[\left(z\mathbf{I} - \mathbf{A}_d\right)^{-1}\left(z\mathbf{B}_{d0} + \mathbf{B}_{d1}\right)\mathbf{u}(z)\right] + \mathbf{D}_d u(z) \\ &= \left[\mathbf{C}_d\left(z\mathbf{I} - \mathbf{A}_d\right)^{-1}\mathbf{B}_{d1} + \mathbf{D}_d\right]\mathbf{u}(z) + z\left[\mathbf{C}_d\left(z\mathbf{I} - \mathbf{A}_d\right)^{-1}B_{d0}\right]\mathbf{u}(z) \end{aligned}$$

Therefore, the transfer function for a system described by Eq. (A.18) is

$$\frac{y(z)}{u(z)} = z\left[\mathbf{C}_d\left(z\mathbf{I} - \mathbf{A}_d\right)^{-1}\mathbf{B}_{d0}\right] + \left[\mathbf{C}_d\left(z\mathbf{I} - \mathbf{A}_d\right)^{-1}\mathbf{B}_{d1} + \mathbf{D}_d\right] \tag{A.19}$$

A.3.1 Example System as in Chapter V. As before, the continuous-time state equations are given in Eq. (A.12). The discrete-time matrices in Eqs. (A.3) and (A.17), for $T = 0.1$, are then given by

$$\mathbf{A}_d = \begin{bmatrix} 0.9930 & 0.0957 \\ -0.1378 & 0.9126 \end{bmatrix}$$

$$\mathbf{B}_{d0} = \begin{bmatrix} 0.0016 \\ 0.0486 \end{bmatrix} \qquad \mathbf{B}_{d1} = \begin{bmatrix} 0.0032 \\ 0.0471 \end{bmatrix}$$

$$\mathbf{C}_d = \begin{bmatrix} 1.44 & 4.8 \end{bmatrix} \qquad \mathbf{D}_d = \begin{bmatrix} 0 \end{bmatrix}$$

Then, following Eq. (A.19), the transfer function for this model is

$$T(z) = \frac{0.2355z^2 + 0.0028z - 0.2244}{z^2 - 1.9056z + 0.9194} \tag{A.20}$$

A.3.2 Further Approximations of the Tustin Transformation. The Tustin transformation assumes the input has the linear property discussed this section, plus it uses the following bilinear approximation for the exponential:

$$e^{\mathbf{A}T} = \left(\mathbf{I} + \frac{\mathbf{A}T}{2}\right)\left(\mathbf{I} - \frac{\mathbf{A}T}{2}\right)^{-1} \tag{A.21}$$

Using this simplification, the discretized matrices in Eqs. (A.3) and (A.17) are

$$\mathbf{A}_d = \begin{bmatrix} 0.9931 & 0.0956 \\ -0.1377 & 0.9128 \end{bmatrix} \qquad \mathbf{B}_{d0} = \mathbf{B}_{d1} = \begin{bmatrix} 0.0024 \\ 0.0478 \end{bmatrix}$$

$$\mathbf{C}_d = \begin{bmatrix} 1.44 & 4.8 \end{bmatrix} \qquad \mathbf{D}_d = \begin{bmatrix} 0 \end{bmatrix}$$

and the Tustin transformed discrete-time transfer function is then given by

$$T(z) = \frac{0.2330z^2 + 0.0069z - 0.2261}{z^2 - 1.9059z + 0.9197} \tag{A.22}$$

A.3.3 Example from Chapter VI. The continuous-time state space equations for this system are obtained from Eq. (6.1):

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} -0.11 & 0.1 \\ 0.2 & -0.202 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 & 0.01 \\ 0.002 & 0 \end{bmatrix} \mathbf{u}(t)$$

$$y(t) = \begin{bmatrix} 0 & -0.1 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0.1 & 0 \end{bmatrix} \mathbf{u}(t)$$

Therefore, the appropriate discretized matrices are

$$\mathbf{A}_d = \begin{bmatrix} 0.9039 & 0.0865 \\ 0.1729 & 0.8244 \end{bmatrix} \qquad \mathbf{C}_d = \begin{bmatrix} 0 & -0.1 \end{bmatrix}$$

$$\mathbf{B}_{d0} = \mathbf{B}_{d1} = 10^{-3} * \begin{bmatrix} 0.0432 & 4.7598 \\ 0.9122 & 0.4323 \end{bmatrix} \qquad \mathbf{D}_d = \begin{bmatrix} 0.1 & 0 \end{bmatrix}$$

and, using Eq. (A.19), the resulting transfer functions are

$$y(z) = \frac{(0.9991z^2 - 1.7284z + 0.7311)10^{-1}}{z^2 - 1.7283z + 0.7302} u_1(z) - \frac{(4.3232z^2 + 8.6464z + 4.3232)10^{-5}}{z^2 - 1.7283z + 0.7302} u_2(z)$$

or, using the form of Eq. (6.9),

$$y(z) = \frac{[(1.7293 - 0.7306)z^2 - 1.7293z + 0.7306]\,10^{-1}}{z^2 - 1.7283z + 0.7302} u_1(z) + \frac{4.3232 * 10^{-5}(z^2 + 2z + 1)}{z^2 - 1.7283z + 0.7302} [u_1(z) - u_2(z)]$$

A.4 Higher Order Assumptions

The constant-input assumption discussed in Section A.2 is perfectly acceptable for systems driven by computer controllers, or other systems where the input to the system actually is held constant over the sample time interval. This assumption, however, is not at all suitable for the identification of continuous-time plants using samples of the continuous-time input and output. For that, one must assume a higher order form on the input.

The linear assumption discussed in Section A.3 produces a relatively good model, provided the sampling rate is high enough to make the input appear somewhat linear between samples. The $e^{\mathbf{A}T}$ approximation made in the Tustin transformation does worsen the system model somewhat, but it allows the simplification of some common transfer function forms. Therefore, one can decrease the number of needed parameters, improving the identification process.

If the sampling rate is too slow, or the initial transient period of the system needs to be used, then the linear/Tustin assumptions must be abandoned for a higher order model.

These higher order models can assume a parabolic or sinusoidal form for the input. Assuming these types of inputs, however, increases the order of the discrete-time model above that of the continuous-time model. This means additional, over-determined, parameters need to be identified. Care must be exercised in this process, because identifying over-determined systems causes many problems.

Appendix B. Noise Plots and Results for Chapter III

This appendix contains plots of representative noise, plots of estimated poles and Bode plots, and numerical results for the frequency domain experiments in Chapter III. Figure B.1 contains representative uncorrelated noise added to the phasor quantities. Figures B.4 and B.5 show the representative uncorrelated noise added to the magnitude and phase angle, and the noise as it appears on the phasor quantities. Figures B.8 and B.9 show the same for the noise added to the magnitude in dB's and phase angle in degrees. The normalized numerical results for each of the examined noise scenarios are given in Tables B.1, B.2, and B.3. Shown are the calculated average error in the estimate $e = \widehat{\theta} - \theta$, the estimation error sigma σ_e, and the algorithm predicted error sigma σ_p for the Monte-Carlo analysis. The n/a entries in the tables indicate that these methods do not produce an estimate for the covariance value. The other figures contain the final estimated Bode plots and poles and zeros for the higher noise cases. Shown are the true and average estimated Bode plots, along with the one sigma bounds on the estimated Bode plots. The pole and zero figures show each of the estimated poles and zeros, along with the true values.

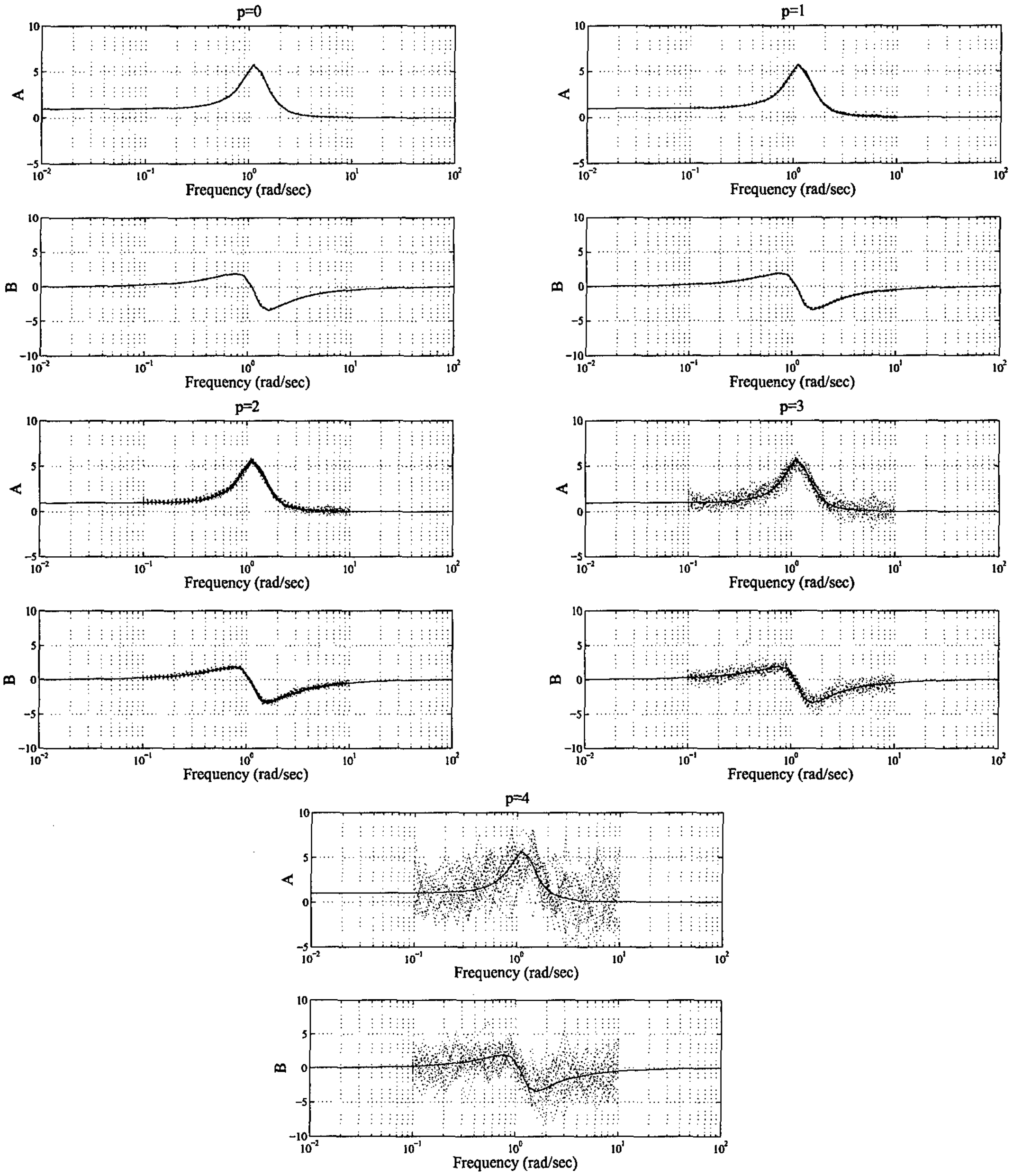


Figure B.1. Noise Added to A and B

Table B.1. Numerical Results for Noise on A and B

Method	θ	e (%)	σ_e	σ_p	e (%)	σ_e	σ_p
		p=0			p=1		
Least Squares	a_1	-15.0562	7.7862	n/a	-65.0425	14.9841	n/a
	a_2	-13.8730	5.7541	n/a	-58.5484	11.4132	n/a
	b_1	-14.3308	6.5080	n/a	-60.8256	12.5748	n/a
	b_2	-14.1483	5.9816	n/a	-59.3727	11.5906	n/a
Generalized Minimum Variance	a_1	-0.0206	0.2690	0.2965	-0.1449	0.8501	0.9362
	a_2	-0.0265	0.1816	0.1764	-0.1530	0.5774	0.5563
	b_1	-0.0017	0.2390	0.2355	-0.0520	0.7538	0.7441
	b_2	-0.0554	0.5991	0.6094	-0.3607	1.8981	1.9235
Generalized Least Squares	a_1	-0.0206	0.2690	n/a	-0.1449	0.8501	n/a
	a_2	-0.0265	0.1816	n/a	-0.1530	0.5774	n/a
	b_1	-0.0017	0.2390	n/a	-0.0520	0.7538	n/a
	b_2	-0.0554	0.5991	n/a	-0.3607	1.8981	n/a
Instrumental Variable	a_1	0.7340	5.3380	11.0606	3.4853	17.2366	11.2673
	a_2	-0.1310	1.4472	6.1274	-0.1748	4.7678	6.1504
	b_1	0.0600	3.2623	6.6856	0.7034	10.4290	6.7548
	b_2	-0.2866	2.3113	6.2029	-0.6365	7.5575	6.2016
		p=2			p=3		
Least Squares	a_1	-100.7956	6.7849	n/a	-102.3768	3.0871	n/a
	a_2	-91.5509	3.8402	n/a	-97.7128	0.7871	n/a
	b_1	-95.0081	4.7130	n/a	-100.4070	1.5802	n/a
	b_2	-92.7353	3.8645	n/a	-99.1122	0.8354	n/a
Generalized Minimum Variance	a_1	-1.2510	2.6522	2.9257	-11.6838	7.1846	8.3129
	a_2	-1.1654	1.8652	1.7240	-9.5793	6.5824	4.5607
	b_1	-0.6375	2.3487	2.3345	-7.3512	6.8212	6.8449
	b_2	-2.9605	6.0049	5.9899	-24.7188	18.1148	16.5515
Generalized Least Squares	a_1	-1.2510	2.6522	n/a	-11.6838	7.1846	n/a
	a_2	-1.1654	1.8652	n/a	-9.5793	6.5824	n/a
	b_1	-0.6375	2.3487	n/a	-7.3512	6.8212	n/a
	b_2	-2.9605	6.0049	n/a	-24.7188	18.1148	n/a
Instrumental Variable	a_1	-73.9084	327.3302	29.3982	-4224.8129	40090.8675	784.6319
	a_2	-21.2100	56.3442	7.3096	127.2910	1885.0227	40.7235
	b_1	-30.0850	117.6643	8.7277	-816.8024	7149.0848	144.3396
	b_2	-21.3158	52.7278	7.3603	426.4470	4774.7141	85.3484

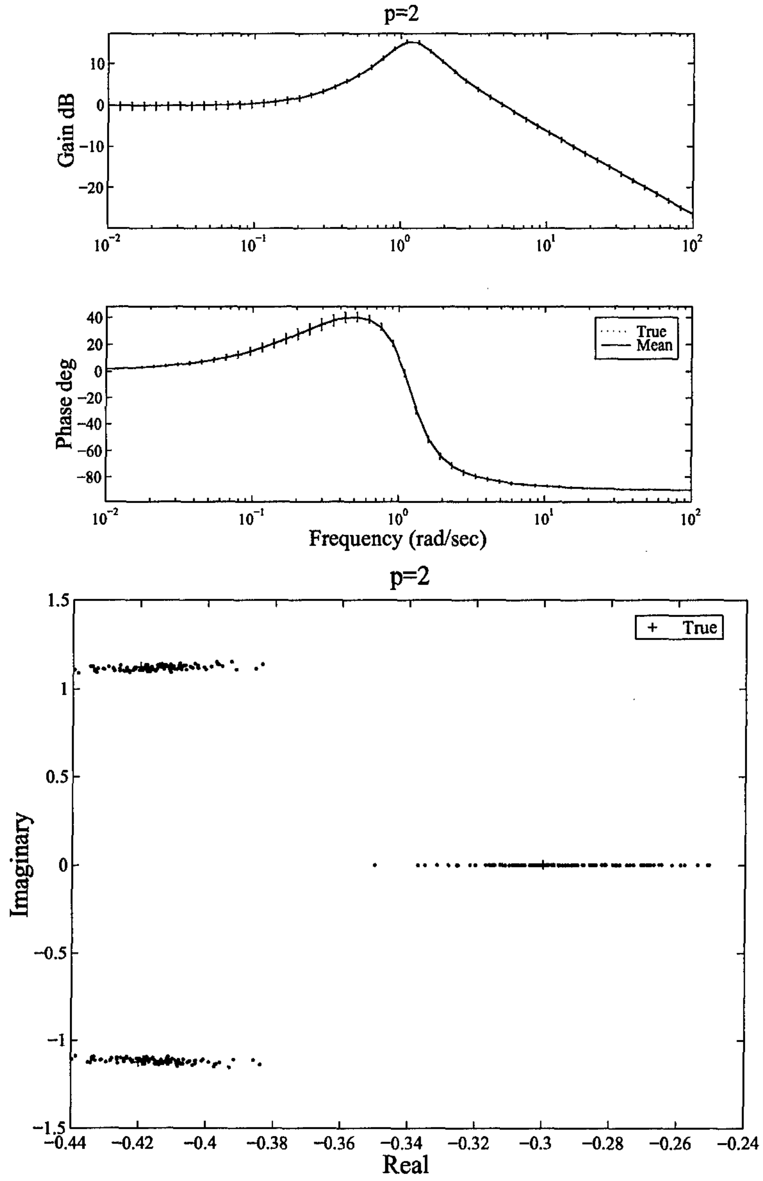


Figure B.2. Final Bode Plots and Roots for A and B Noise - $p = 2$

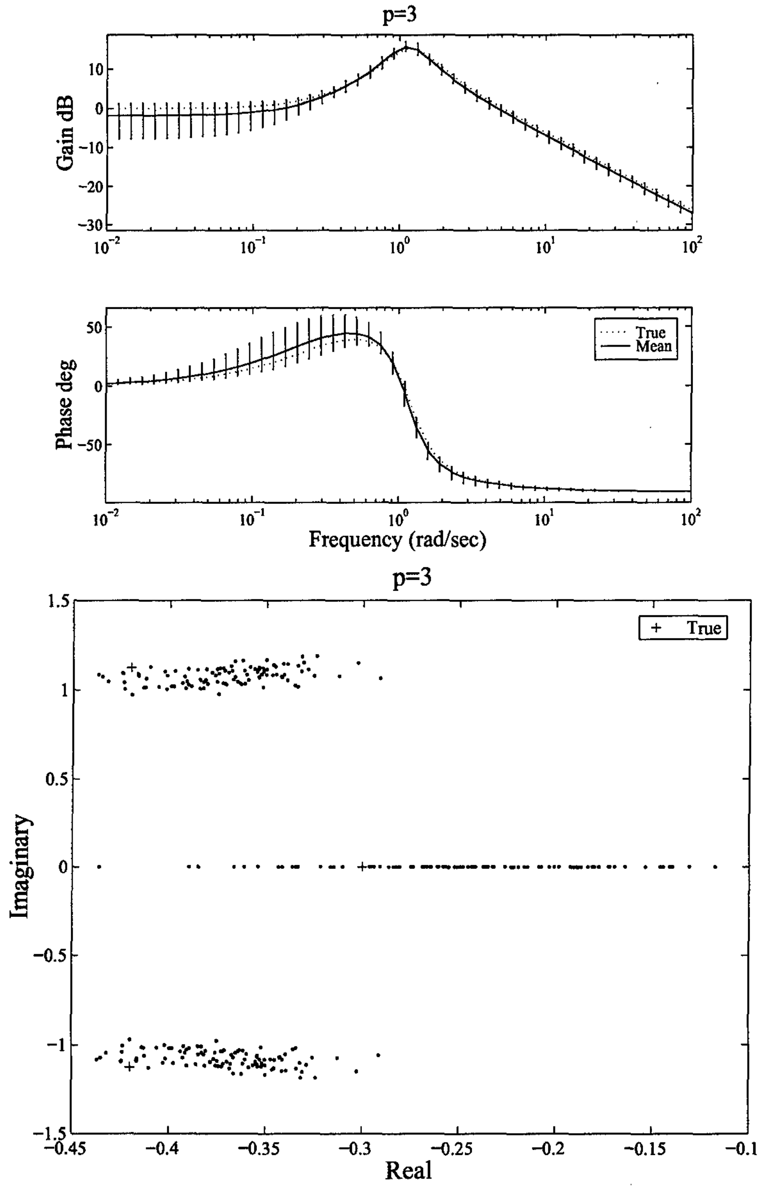


Figure B.3. Final Bode Plots and Roots for A and B Noise - $p = 3$

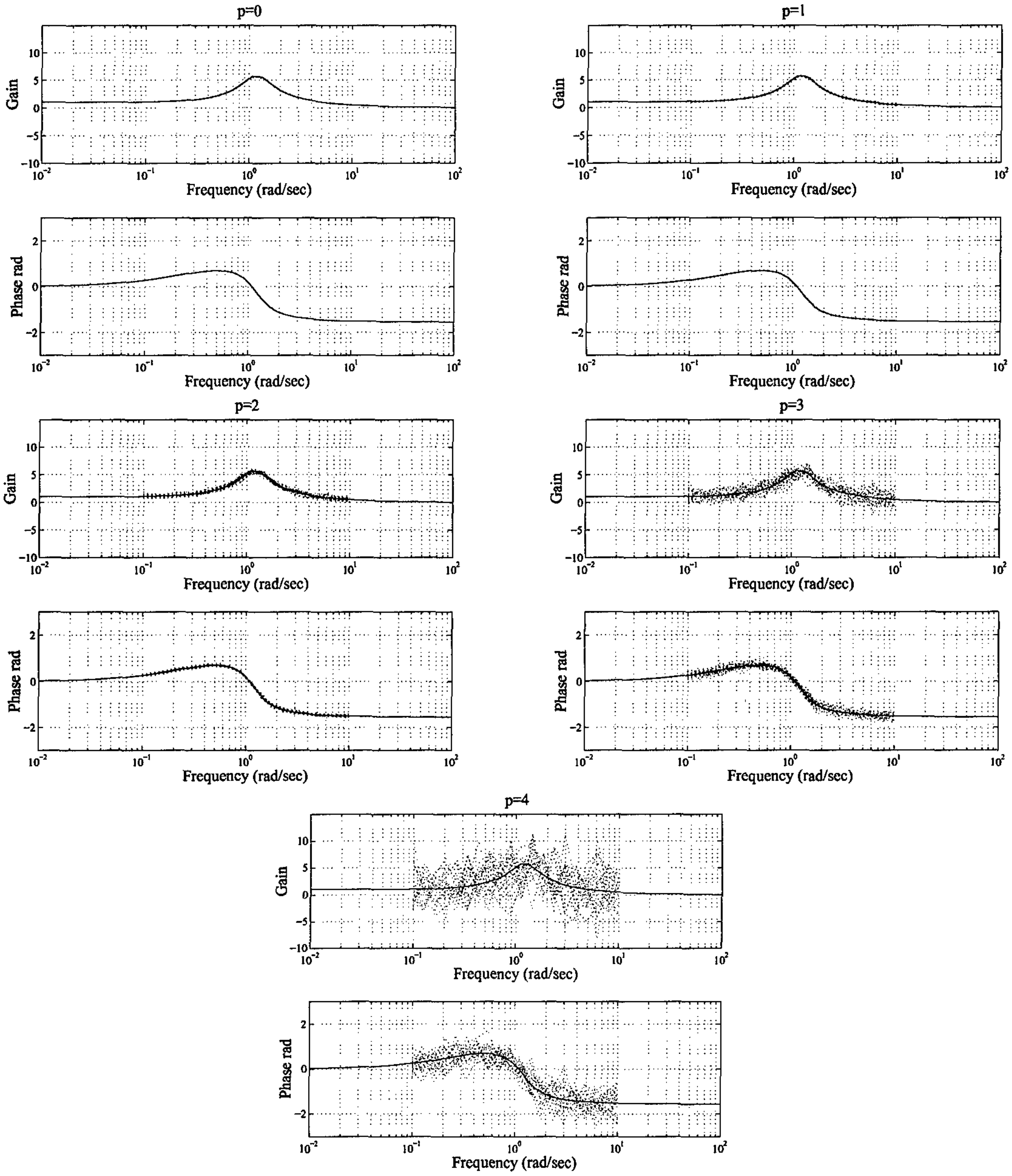


Figure B.4. Noise Added to Magnitude and Radian Phase

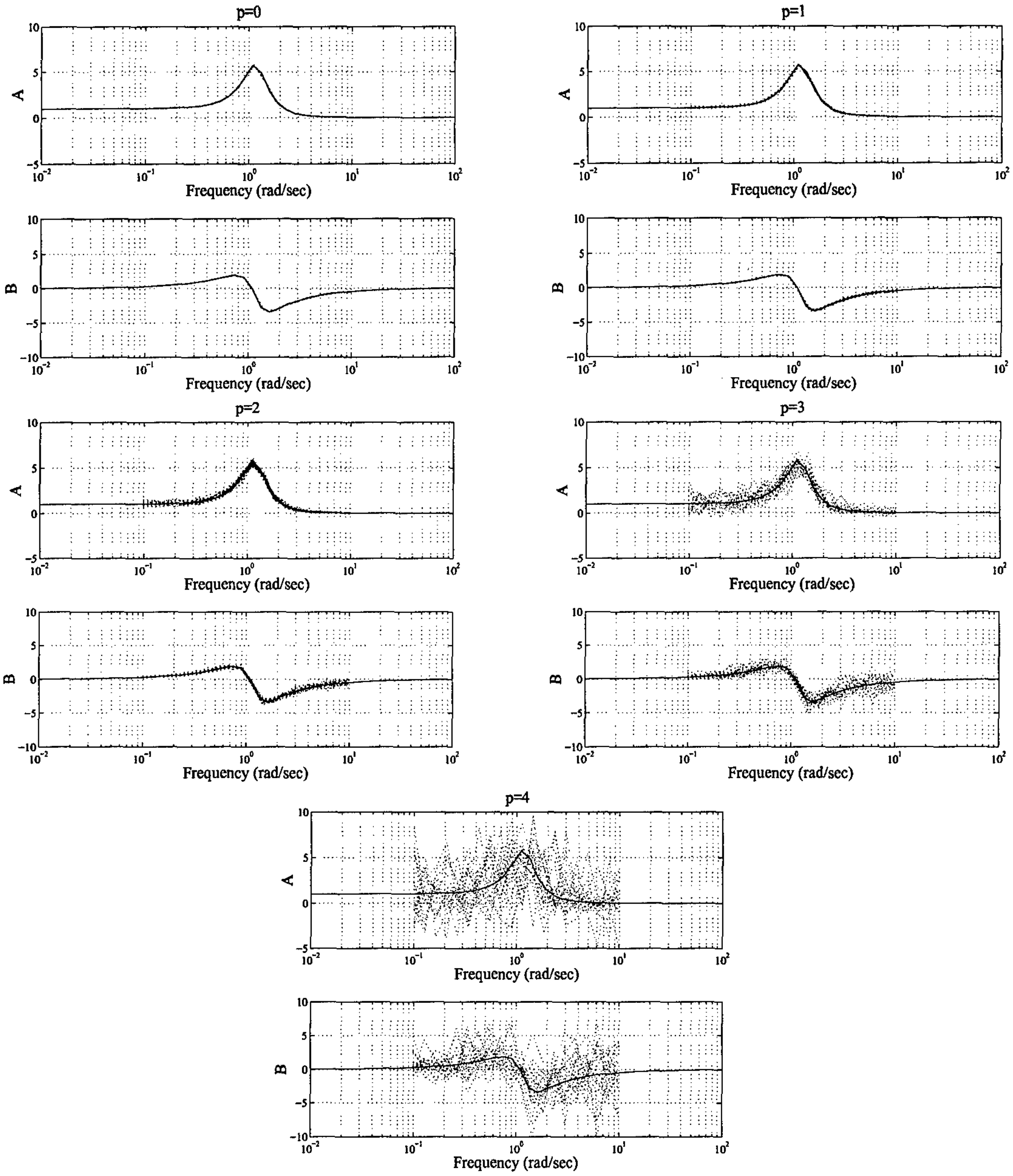


Figure B.5. Transformed Noise On Magnitude and Radian Phase

Table B.2. Numerical Results for Noise on R and ϕ

Method	θ	e (%)	σ_e	σ_p	e (%)	σ_e	σ_p
		p=0			p=1		
Least Squares	a_1	-13.7859	8.3089	n/a	-60.8193	17.0990	n/a
	a_2	-12.7156	6.7777	n/a	-54.5728	14.5087	n/a
	b_1	-13.2254	7.1137	n/a	-56.8998	15.1957	n/a
	b_2	-13.0119	7.0774	n/a	-55.4231	14.8450	n/a
Generalized Minimum Variance	a_1	-0.0333	0.2404	0.2465	-0.1644	0.7582	0.7779
	a_2	-0.0214	0.1388	0.1509	-0.1232	0.4388	0.4763
	b_1	-0.0191	0.2253	0.2433	-0.1101	0.7114	0.7685
	b_2	-0.0253	0.3002	0.3322	-0.1700	0.9492	1.0490
Generalized Least Squares	a_1	-0.0327	0.3053	n/a	-0.2092	0.9694	n/a
	a_2	-0.0332	0.1799	n/a	-0.1760	0.5730	n/a
	b_1	-0.0150	0.2670	n/a	-0.1166	0.8457	n/a
	b_2	-0.0810	0.5487	n/a	-0.4561	1.7458	n/a
Instrumental Variable	a_1	0.5676	3.6425	11.0571	2.3534	11.3904	11.2125
	a_2	-0.2263	1.8593	6.1239	-0.2770	6.4606	6.1379
	b_1	-0.1548	1.9778	6.6750	0.0437	6.8177	6.7071
	b_2	-0.4102	3.2066	6.1972	-0.6982	10.8584	6.1859
		p=2			p=3		
Least Squares	a_1	-99.4138	7.4107	n/a	-102.6223	2.1372	n/a
	a_2	-89.8717	5.4065	n/a	-97.3743	1.1510	n/a
	b_1	-93.4695	5.8279	n/a	-100.2593	1.3004	n/a
	b_2	-91.1127	5.5227	n/a	-98.8350	1.2308	n/a
Generalized Minimum Variance	a_1	-1.0872	2.4214	2.4171	-7.6534	9.6950	6.5178
	a_2	-0.9521	1.4101	1.4821	-9.7354	6.5211	4.0128
	b_1	-0.8351	2.2562	2.4079	-7.3839	7.9341	7.0072
	b_2	-1.4429	3.0272	3.2757	-14.5805	11.1922	9.0866
Generalized Least Squares	a_1	-1.7115	3.0686	n/a	-83.8464	35.6204	n/a
	a_2	-1.2550	1.8752	n/a	-78.8845	35.7437	n/a
	b_1	-1.0672	2.6737	n/a	-81.7519	37.8196	n/a
	b_2	-3.3997	5.6253	n/a	-83.9706	32.0320	n/a
Instrumental Variable	a_1	-46.5619	143.6145	20.0687	-187.3172	387.1076	40.0542
	a_2	-16.0451	39.5066	6.7231	-65.2617	65.3378	7.0002
	b_1	-23.7460	58.3411	7.0748	-93.5016	106.3434	8.7700
	b_2	-16.9293	43.3818	6.4407	-63.4651	80.3390	7.7578

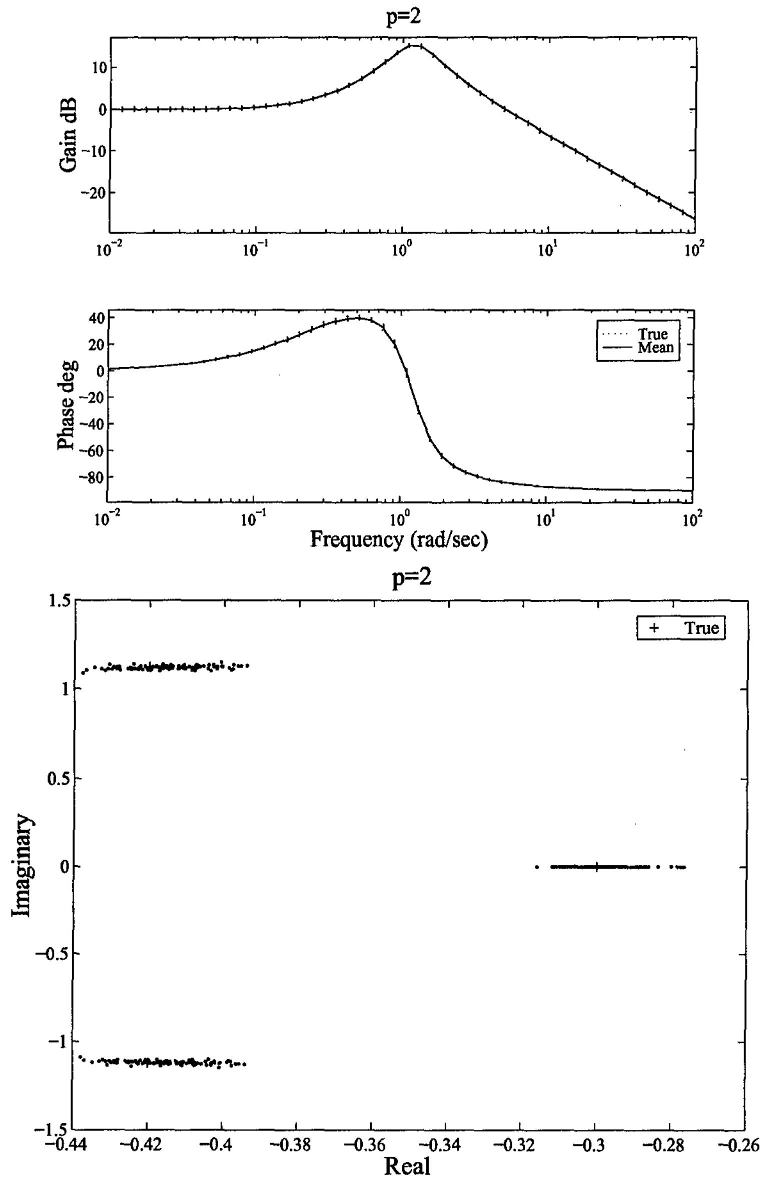


Figure B.6. Final Bode Plots and Roots for M and ϕ Noise - $p = 2$

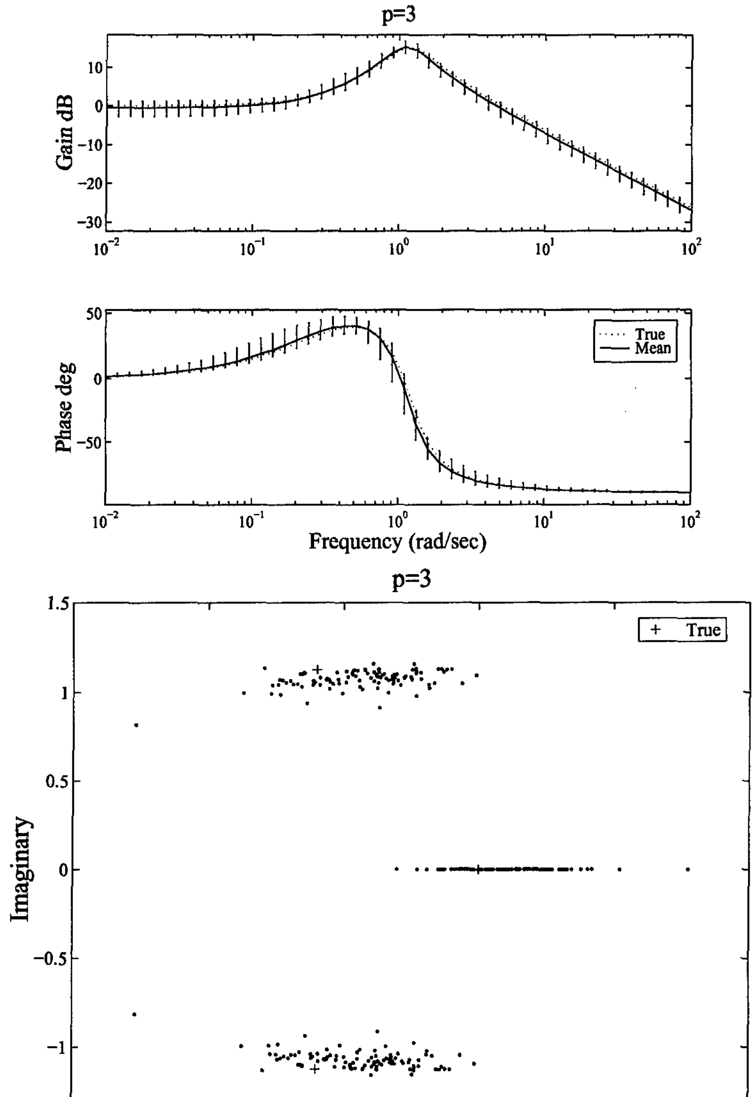


Figure B.7. Final Bode Plots and Roots for M and ϕ Noise - $p = 3$

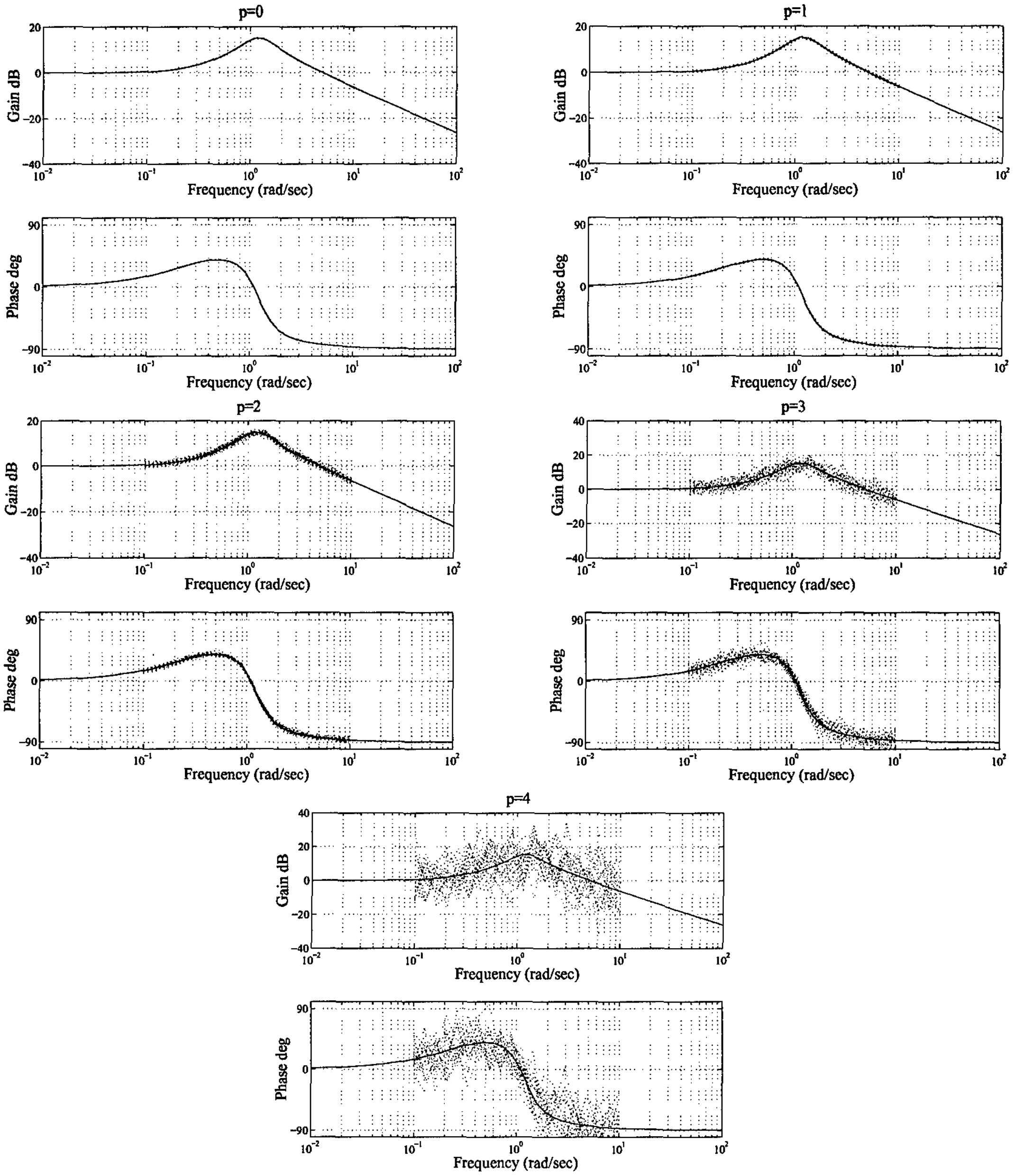
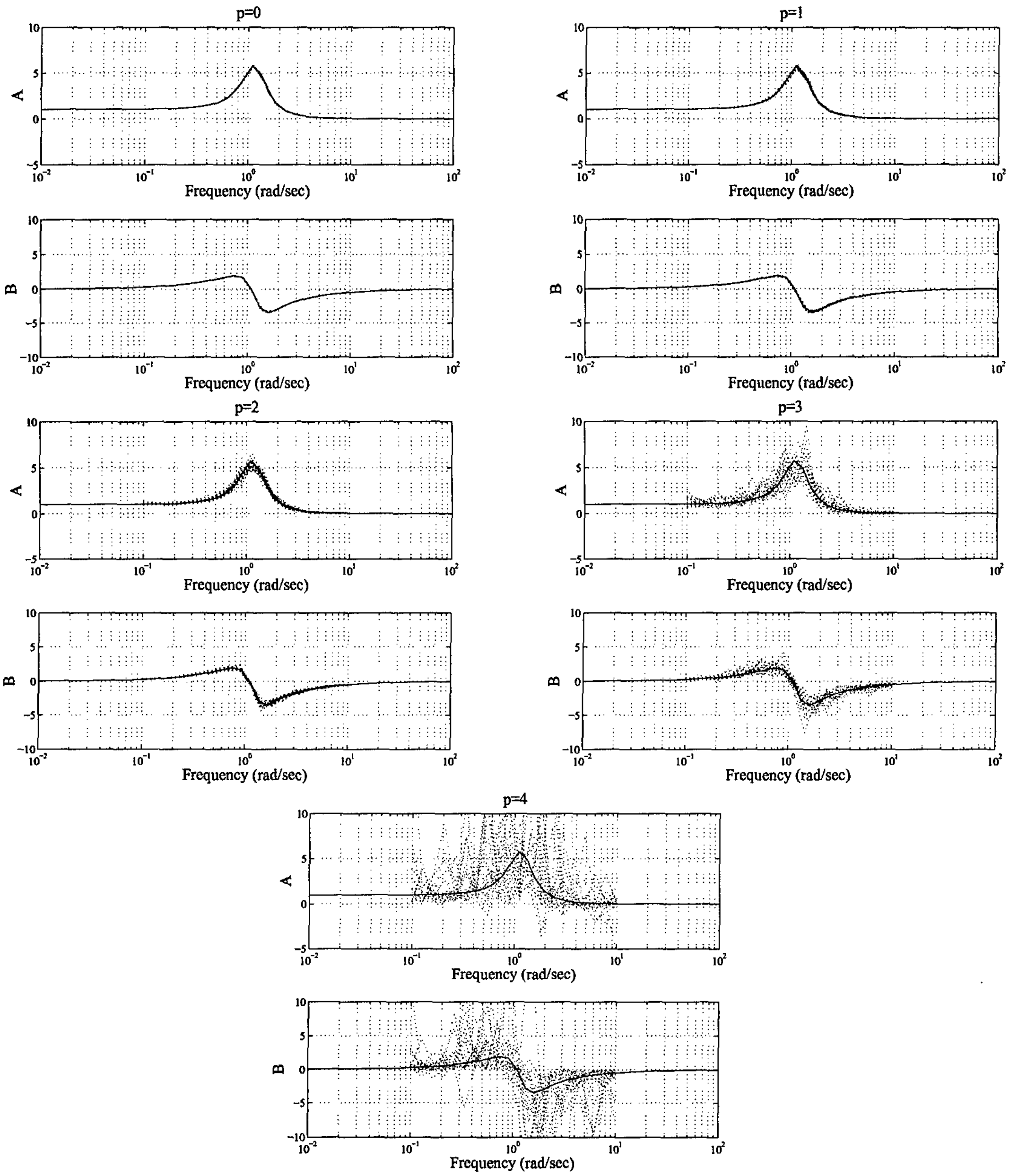


Figure B.8. Noise Added to dB Magnitude and Degree Phase


p=0

p=1

p=2

p=3

p=4

Figure B.9. Transformed Noise on dB Magnitude and Degree Phase

Table B.3. Numerical Results for Noise on R_{bB} and ϕ_{deg}

Method	θ	e (%)	σ_e	σ_p	e (%)	σ_e	σ_p
		p=0			p=1		
Least Squares	a_1	-2.9445	2.4267	n/a	-23.3012	10.8525	n/a
	a_2	-2.8304	1.6040	n/a	-21.0595	8.6073	n/a
	b_1	-2.9268	1.7780	n/a	-21.9066	9.1428	n/a
	b_2	-2.9272	1.8480	n/a	-21.4273	8.8081	n/a
Generalized Minimum Variance	a_1	-0.0273	0.2526	0.2501	-0.1481	0.8007	0.7902
	a_2	-0.0190	0.1429	0.1546	-0.1174	0.4508	0.4876
	b_1	-0.0040	0.1951	0.2044	-0.0010	0.6165	0.6463
	b_2	-0.0137	0.2613	0.3055	-0.0732	0.8231	0.9650
Generalized Least Squares	a_1	-0.0814	0.4701	n/a	-0.4840	1.5197	n/a
	a_2	-0.0368	0.2525	n/a	-0.1567	0.8002	n/a
	b_1	-0.0293	0.2936	n/a	-0.1965	0.9365	n/a
	b_2	-0.0719	0.6230	n/a	-0.3082	1.9785	n/a
Instrumental Variable	a_1	0.1858	1.7507	11.0242	0.7129	5.5303	11.0621
	a_2	-0.1124	0.7662	6.1226	-0.2948	2.4593	6.1177
	b_1	-0.0802	0.9566	6.6708	-0.1318	3.0493	6.6748
	b_2	-0.1836	1.2616	6.2037	-0.4647	4.0419	6.1913
		p=2			p=3		
Least Squares	a_1	-76.6357	14.0096	n/a	-100.9812	5.0805	n/a
	a_2	-68.3917	11.0486	n/a	-92.4881	3.6376	n/a
	b_1	-71.2315	11.7539	n/a	-95.8282	4.0113	n/a
	b_2	-69.2936	11.1049	n/a	-93.6736	3.5043	n/a
Generalized Minimum Variance	a_1	-1.0841	2.5686	2.4737	-9.1244	8.9164	7.0716
	a_2	-0.9353	1.4612	1.5155	-8.1383	6.8051	4.1673
	b_1	0.1100	1.9513	2.0448	1.4464	6.5124	6.5088
	b_2	-0.5225	2.6053	3.0288	-4.2140	9.7562	9.0579
Generalized Least Squares	a_1	-3.7523	5.2088	n/a	-40.9184	28.3573	n/a
	a_2	-0.8867	2.6295	n/a	-14.6615	28.0427	n/a
	b_1	-1.6765	3.1116	n/a	-27.5110	29.5469	n/a
	b_2	-1.7881	6.5015	n/a	-24.0412	37.7590	n/a
Instrumental Variable	a_1	1.6252	28.1418	11.2921	21.5836	1089.3221	42.1069
	a_2	-2.3172	18.1915	6.1431	-29.5097	81.5201	7.5469
	b_1	-1.3839	21.0757	6.7647	-12.3400	236.9992	10.8943
	b_2	-2.6517	21.6585	6.2209	-35.4831	165.4659	9.4785

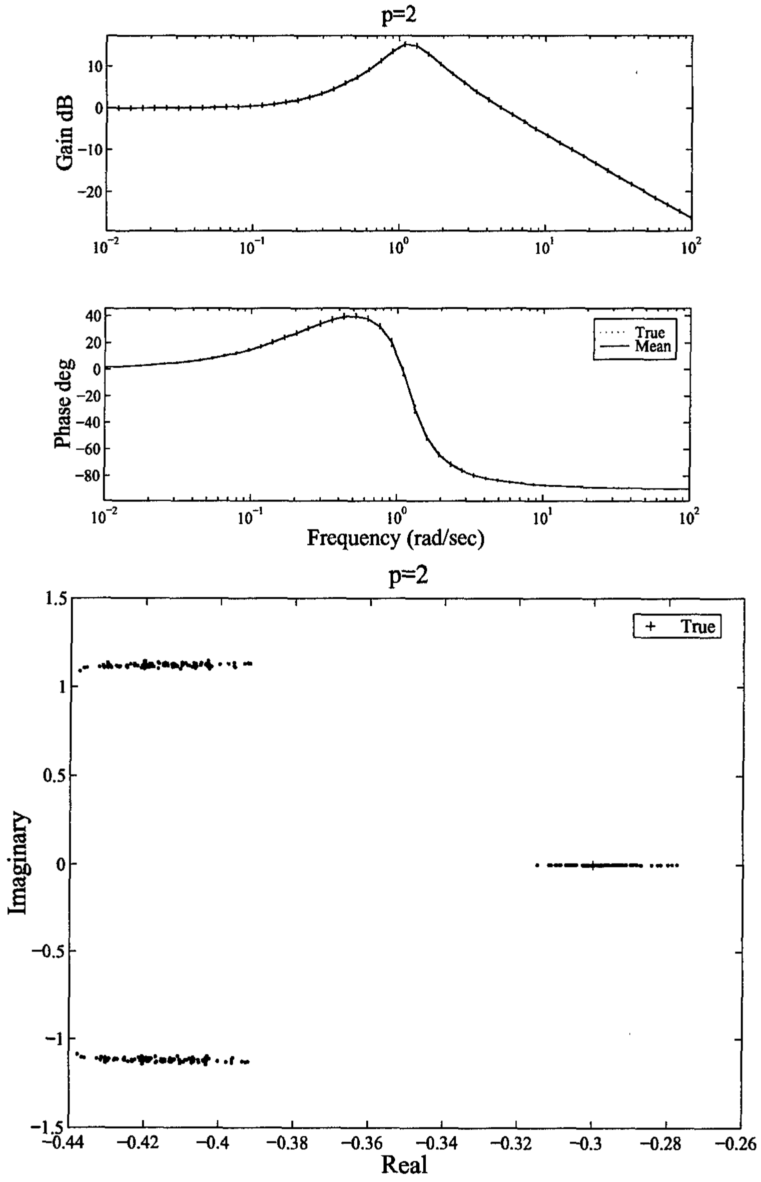


Figure B.10. Final Bode Plots and Roots for M_{dB} and ϕ_{deg} Noise - $p = 2$

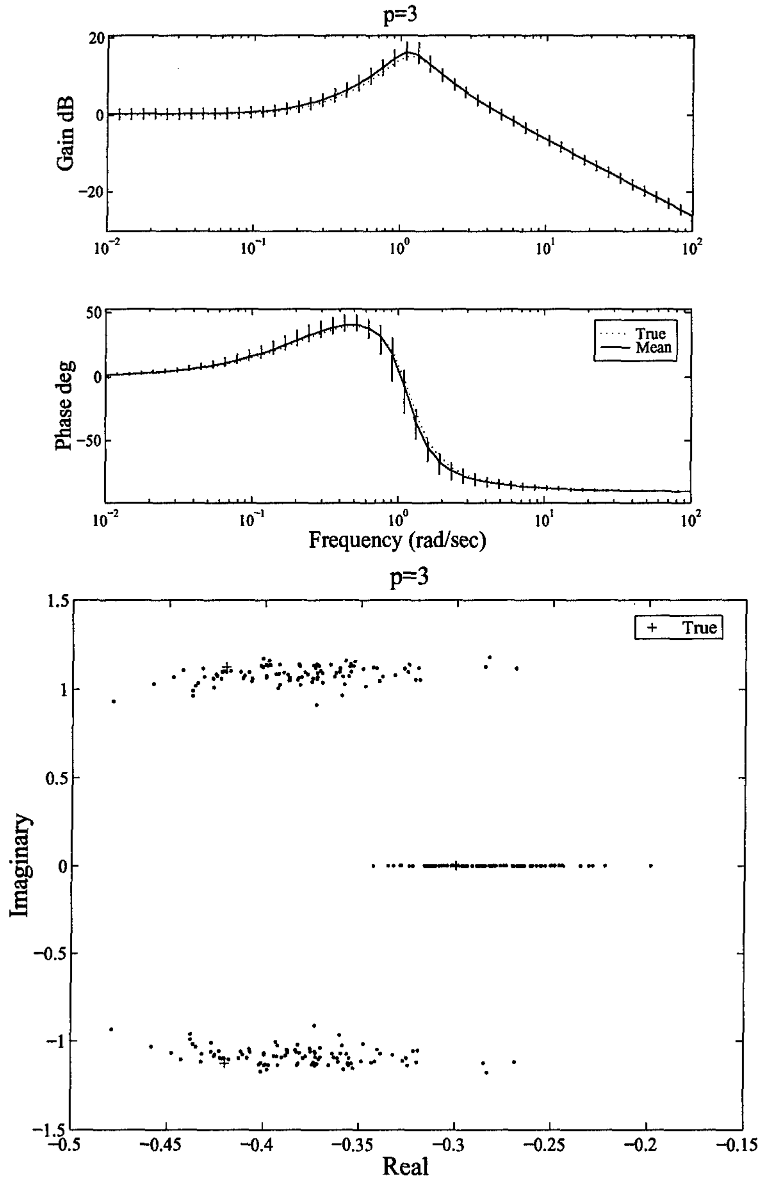


Figure B.11. Final Bode Plots and Roots for M_{dB} and ϕ_{deg} Noise - $p = 3$

Appendix C. Numerical Results for Chapter IV

This appendix contains the numerical results for the unmodeled dynamics experiments performed in Chapter IV. Given are the estimated parameters and the corresponding poles and zero of the estimated plant for each of the measurement windows. Table C.1 contains the estimated parameters for the baseline second-order plant. Tables C.2 and C.3 contain the estimated parameters for the fourth-order plants containing the low and high frequency dynamics, respectively. Table C.4 contains the estimated parameters for the sixth-order plant. Finally, Table C.5 contains the poles and zero for each of the measurement windows in each of the cases.

Table C.1. Estimation Results Using Only the Second-Order Plant

Run	$\widehat{\theta} - \theta$	σ_e	σ_p	$\widehat{\theta} - \theta$	σ_e	σ_p
	$\theta = a1 = -0.84$			$\theta = a2 = -1.44$		
1	$1.20\ 10^{0}$	$1.07\ 10^{-2}$	$1.25\ 10^{-2}$	$1.44\ 10^{0}$	$1.72\ 10^{-3}$	$6.43\ 10^{-4}$
2	$1.20\ 10^{0}$	$9.82\ 10^{-3}$	$7.95\ 10^{-3}$	$1.44\ 10^{0}$	$4.20\ 10^{-3}$	$1.26\ 10^{-3}$
3	$1.22\ 10^{0}$	$8.93\ 10^{-3}$	$4.77\ 10^{-3}$	$1.44\ 10^{0}$	$5.34\ 10^{-3}$	$2.28\ 10^{-3}$
4	$1.20\ 10^{0}$	$1.25\ 10^{-1}$	$9.48\ 10^{-3}$	$1.38\ 10^{0}$	$6.73\ 10^{-2}$	$8.98\ 10^{-3}$
5	$4.77\ 10^{-1}$	$4.79\ 10^{-2}$	$6.94\ 10^{-2}$	$6.66\ 10^{-1}$	$7.08\ 10^{-2}$	$9.44\ 10^{-2}$
6	$7.66\ 10^{-2}$	$5.81\ 10^{-2}$	$4.97\ 10^{-2}$	$9.59\ 10^{-2}$	$7.38\ 10^{-2}$	$6.15\ 10^{-2}$
7	$-1.08\ 10^{-3}$	$9.95\ 10^{-3}$	$1.07\ 10^{-2}$	$-6.00\ 10^{-4}$	$7.46\ 10^{-3}$	$7.01\ 10^{-3}$
8	$1.08\ 10^{-3}$	$2.37\ 10^{-3}$	$2.70\ 10^{-3}$	$-6.86\ 10^{-4}$	$2.08\ 10^{-3}$	$2.98\ 10^{-3}$
9	$2.15\ 10^{-3}$	$4.08\ 10^{-3}$	$3.65\ 10^{-3}$	$2.73\ 10^{-3}$	$5.75\ 10^{-3}$	$6.29\ 10^{-3}$
10	$7.26\ 10^{-3}$	$1.98\ 10^{-2}$	$2.45\ 10^{-2}$	$-7.60\ 10^{-4}$	$1.30\ 10^{-2}$	$1.59\ 10^{-2}$
11	$4.20\ 10^{-2}$	$1.52\ 10^{-1}$	$1.46\ 10^{-1}$	$3.47\ 10^{-2}$	$8.99\ 10^{-2}$	$1.03\ 10^{-1}$
12	$-2.95\ 10^{-2}$	$8.02\ 10^{-1}$	$5.88\ 10^{-1}$	$-6.64\ 10^{-2}$	$5.19\ 10^{-1}$	$4.05\ 10^{-1}$
13	$-4.26\ 10^{-1}$	$2.67\ 10^{0}$	$1.61\ 10^{0}$	$3.98\ 10^{-3}$	$1.86\ 10^{0}$	$1.32\ 10^{0}$
14	$-7.75\ 10^{-1}$	$5.84\ 10^{0}$	$3.05\ 10^{0}$	$-9.35\ 10^{-3}$	$3.71\ 10^{0}$	$3.38\ 10^{0}$
15	$2.39\ 10^{-1}$	$9.67\ 10^{0}$	$5.02\ 10^{0}$	$-3.14\ 10^{0}$	$1.48\ 10^{1}$	$1.04\ 10^{1}$
	$\theta = b1 = 4.8$			$\theta = b2 = 1.44$		
1	$-5.16\ 10^{0}$	$9.94\ 10^{-3}$	$1.27\ 10^{-2}$	$-1.44\ 10^{0}$	$1.73\ 10^{-3}$	$6.43\ 10^{-4}$
2	$-5.16\ 10^{0}$	$9.00\ 10^{-3}$	$9.23\ 10^{-3}$	$-1.44\ 10^{0}$	$4.23\ 10^{-3}$	$1.26\ 10^{-3}$
3	$-5.18\ 10^{0}$	$7.22\ 10^{-3}$	$9.35\ 10^{-3}$	$-1.44\ 10^{0}$	$5.47\ 10^{-3}$	$2.29\ 10^{-3}$
4	$-5.05\ 10^{0}$	$2.92\ 10^{-1}$	$3.26\ 10^{-2}$	$-1.38\ 10^{0}$	$6.59\ 10^{-2}$	$9.00\ 10^{-3}$
5	$-2.38\ 10^{0}$	$2.51\ 10^{-1}$	$3.37\ 10^{-1}$	$-6.79\ 10^{-1}$	$7.33\ 10^{-2}$	$9.61\ 10^{-2}$
6	$-3.71\ 10^{-1}$	$2.87\ 10^{-1}$	$2.38\ 10^{-1}$	$-1.01\ 10^{-1}$	$7.74\ 10^{-2}$	$6.48\ 10^{-2}$
7	$2.35\ 10^{-3}$	$4.82\ 10^{-2}$	$4.48\ 10^{-2}$	$-9.42\ 10^{-4}$	$1.05\ 10^{-2}$	$9.96\ 10^{-3}$
8	$-5.17\ 10^{-3}$	$1.29\ 10^{-2}$	$1.70\ 10^{-2}$	$1.18\ 10^{-3}$	$8.79\ 10^{-3}$	$1.11\ 10^{-2}$
9	$-2.60\ 10^{-3}$	$1.49\ 10^{-2}$	$1.55\ 10^{-2}$	$-1.84\ 10^{-2}$	$3.36\ 10^{-2}$	$3.29\ 10^{-2}$
10	$-6.96\ 10^{-3}$	$2.24\ 10^{-2}$	$2.02\ 10^{-2}$	$-4.22\ 10^{-2}$	$1.11\ 10^{-1}$	$1.43\ 10^{-1}$
11	$-6.09\ 10^{-4}$	$2.28\ 10^{-2}$	$2.34\ 10^{-2}$	$-2.21\ 10^{-1}$	$7.79\ 10^{-1}$	$7.49\ 10^{-1}$
12	$-4.82\ 10^{-3}$	$1.99\ 10^{-2}$	$2.39\ 10^{-2}$	$1.49\ 10^{-1}$	$3.95\ 10^{0}$	$2.89\ 10^{0}$
13	$5.72\ 10^{-4}$	$2.43\ 10^{-2}$	$2.37\ 10^{-2}$	$2.08\ 10^{0}$	$1.28\ 10^{1}$	$7.80\ 10^{0}$
14	$-5.30\ 10^{-3}$	$2.04\ 10^{-2}$	$2.37\ 10^{-2}$	$3.77\ 10^{0}$	$2.81\ 10^{1}$	$1.47\ 10^{1}$
15	$-1.11\ 10^{-2}$	$3.02\ 10^{-2}$	$2.38\ 10^{-2}$	$-1.04\ 10^{0}$	$4.65\ 10^{1}$	$2.41\ 10^{1}$

Table C.2. Estimation Results With the Low Frequency Mode Addition

Run	$\widehat{\theta}-\theta$	σ_e	σ_p	$\widehat{\theta}-\theta$	σ_e	σ_p
	$\theta = a1 = -0.84$			$\theta = a2 = -1.44$		
1	8.59 10^{-1}	2.14 10^{-4}	1.73 10^{-4}	1.44 10^{0}	2.84 10^{-6}	3.31 10^{-6}
2	8.56 10^{-1}	1.30 10^{-4}	1.06 10^{-4}	1.44 10^{0}	6.08 10^{-6}	3.22 10^{-6}
3	8.38 10^{-1}	3.75 10^{-5}	1.91 10^{-5}	1.43 10^{0}	6.35 10^{-7}	7.31 10^{-7}
4	8.34 10^{-1}	3.45 10^{-5}	2.99 10^{-5}	1.43 10^{0}	1.28 10^{-6}	1.90 10^{-6}
5	1.22 10^{0}	6.46 10^{-3}	3.12 10^{-3}	1.45 10^{0}	1.63 10^{-3}	5.93 10^{-4}
6	3.66 10^{-1}	4.99 10^{-2}	4.07 10^{-2}	3.38 10^{-1}	7.42 10^{-2}	5.71 10^{-2}
7	-2.75 10^{-4}	1.01 10^{-2}	1.08 10^{-2}	-8.34 10^{-3}	7.67 10^{-3}	7.15 10^{-3}
8	5.85 10^{-4}	2.37 10^{-3}	2.70 10^{-3}	-1.80 10^{-4}	2.08 10^{-3}	2.99 10^{-3}
9	2.90 10^{-3}	4.08 10^{-3}	3.66 10^{-3}	5.28 10^{-3}	5.75 10^{-3}	6.28 10^{-3}
10	9.62 10^{-3}	1.98 10^{-2}	2.44 10^{-2}	1.33 10^{-3}	1.30 10^{-2}	1.59 10^{-2}
11	4.46 10^{-2}	1.52 10^{-1}	1.46 10^{-1}	3.59 10^{-2}	9.00 10^{-2}	1.03 10^{-1}
12	-2.75 10^{-2}	8.01 10^{-1}	5.88 10^{-1}	-6.60 10^{-2}	5.20 10^{-1}	4.06 10^{-1}
13	-4.28 10^{-1}	2.66 10^{0}	1.61 10^{0}	1.78 10^{-3}	1.86 10^{0}	1.32 10^{0}
14	-7.77 10^{-1}	5.84 10^{0}	3.05 10^{0}	-1.25 10^{-2}	3.72 10^{0}	3.39 10^{0}
15	2.37 10^{-1}	9.67 10^{0}	5.02 10^{0}	-3.14 10^{0}	1.48 10^{1}	1.04 10^{1}
	$\theta = b1 = 4.8$			$\theta = b2 = 1.44$		
1	-4.80 10^{0}	2.71 10^{-5}	2.47 10^{-5}	-1.44 10^{0}	1.15 10^{-7}	1.29 10^{-7}
2	-4.79 10^{0}	4.83 10^{-5}	4.63 10^{-5}	-1.44 10^{0}	1.84 10^{-6}	8.83 10^{-7}
3	-4.79 10^{0}	4.74 10^{-5}	6.97 10^{-5}	-1.44 10^{0}	1.27 10^{-5}	5.86 10^{-6}
4	-4.80 10^{0}	8.87 10^{-5}	1.09 10^{-4}	-1.44 10^{0}	2.53 10^{-5}	1.83 10^{-5}
5	-5.48 10^{0}	5.92 10^{-3}	4.34 10^{-3}	-1.46 10^{0}	2.91 10^{-3}	1.04 10^{-3}
6	-1.35 10^{0}	2.84 10^{-1}	2.21 10^{-1}	-2.52 10^{-1}	8.67 10^{-2}	6.62 10^{-2}
7	5.21 10^{-2}	4.96 10^{-2}	4.60 10^{-2}	3.75 10^{-2}	1.10 10^{-2}	1.03 10^{-2}
8	-2.11 10^{-3}	1.29 10^{-2}	1.70 10^{-2}	9.57 10^{-4}	8.78 10^{-3}	1.11 10^{-2}
9	-1.37 10^{-2}	1.49 10^{-2}	1.55 10^{-2}	-3.46 10^{-2}	3.35 10^{-2}	3.28 10^{-2}
10	-2.21 10^{-2}	2.24 10^{-2}	2.02 10^{-2}	-6.72 10^{-2}	1.11 10^{-1}	1.42 10^{-1}
11	-1.63 10^{-2}	2.27 10^{-2}	2.34 10^{-2}	-2.46 10^{-1}	7.75 10^{-1}	7.45 10^{-1}
12	-2.05 10^{-2}	1.98 10^{-2}	2.38 10^{-2}	1.26 10^{-1}	3.93 10^{0}	2.88 10^{0}
13	-1.51 10^{-2}	2.42 10^{-2}	2.37 10^{-2}	2.06 10^{0}	1.28 10^{1}	7.77 10^{0}
14	-2.10 10^{-2}	2.04 10^{-2}	2.36 10^{-2}	3.76 10^{0}	2.80 10^{1}	1.46 10^{1}
15	-2.68 10^{-2}	3.01 10^{-2}	2.38 10^{-2}	-1.05 10^{0}	4.64 10^{1}	2.40 10^{1}

Table C.3. Estimation Results With the High Frequency Mode Addition

Run	$\widehat{\theta}-\theta$	σ_e	σ_p	$\widehat{\theta}-\theta$	σ_e	σ_p
	$\theta = a1 = -0.84$			$\theta = a2 = -1.44$		
1	$1.20\ 10^{0}$	$1.06\ 10^{-2}$	$1.23\ 10^{-2}$	$1.44\ 10^{0}$	$1.71\ 10^{-3}$	$6.39\ 10^{-4}$
2	$1.20\ 10^{0}$	$9.74\ 10^{-3}$	$7.84\ 10^{-3}$	$1.44\ 10^{0}$	$4.18\ 10^{-3}$	$1.25\ 10^{-3}$
3	$1.21\ 10^{0}$	$8.96\ 10^{-3}$	$4.72\ 10^{-3}$	$1.44\ 10^{0}$	$5.33\ 10^{-3}$	$2.27\ 10^{-3}$
4	$1.20\ 10^{0}$	$1.26\ 10^{-1}$	$9.51\ 10^{-3}$	$1.38\ 10^{0}$	$6.80\ 10^{-2}$	$9.01\ 10^{-3}$
5	$4.74\ 10^{-1}$	$4.82\ 10^{-2}$	$6.97\ 10^{-2}$	$6.66\ 10^{-1}$	$7.10\ 10^{-2}$	$9.44\ 10^{-2}$
6	$7.28\ 10^{-2}$	$5.82\ 10^{-2}$	$4.98\ 10^{-2}$	$9.69\ 10^{-2}$	$7.36\ 10^{-2}$	$6.14\ 10^{-2}$
7	$-4.30\ 10^{-3}$	$9.97\ 10^{-3}$	$1.07\ 10^{-2}$	$-5.53\ 10^{-5}$	$7.45\ 10^{-3}$	$7.01\ 10^{-3}$
8	$1.04\ 10^{-3}$	$2.37\ 10^{-3}$	$2.70\ 10^{-3}$	$-6.10\ 10^{-4}$	$2.08\ 10^{-3}$	$2.98\ 10^{-3}$
9	$6.13\ 10^{-4}$	$4.04\ 10^{-3}$	$3.64\ 10^{-3}$	$-9.57\ 10^{-3}$	$5.74\ 10^{-3}$	$6.30\ 10^{-3}$
10	$-5.64\ 10^{-2}$	$2.16\ 10^{-2}$	$2.54\ 10^{-2}$	$-7.39\ 10^{-2}$	$1.35\ 10^{-2}$	$1.59\ 10^{-2}$
11	$-1.05\ 10^{0}$	$2.82\ 10^{-1}$	$1.77\ 10^{-1}$	$-8.38\ 10^{-1}$	$1.14\ 10^{-1}$	$8.86\ 10^{-2}$
12	$-9.08\ 10^{0}$	$9.90\ 10^{-1}$	$6.67\ 10^{-1}$	$-7.38\ 10^{0}$	$5.84\ 10^{-1}$	$3.07\ 10^{-1}$
13	$-5.74\ 10^{0}$	$1.37\ 10^{-1}$	$5.64\ 10^{-2}$	$-8.95\ 10^{1}$	$2.84\ 10^{-1}$	$2.91\ 10^{-1}$
14	$-6.65\ 10^{-1}$	$1.12\ 10^{-1}$	$4.20\ 10^{-2}$	$-2.77\ 10^{2}$	$7.10\ 10^{-1}$	$6.96\ 10^{-1}$
15	$-1.44\ 10^{1}$	$4.03\ 10^{-1}$	$4.47\ 10^{-1}$	$-5.35\ 10^{2}$	$5.11\ 10^{0}$	$4.11\ 10^{0}$
	$\theta = b1 = 4.8$			$\theta = b2 = 1.44$		
1	$-5.15\ 10^{0}$	$9.74\ 10^{-3}$	$1.25\ 10^{-2}$	$-1.44\ 10^{0}$	$1.71\ 10^{-3}$	$6.36\ 10^{-4}$
2	$-5.16\ 10^{0}$	$8.90\ 10^{-3}$	$9.09\ 10^{-3}$	$-1.44\ 10^{0}$	$4.19\ 10^{-3}$	$1.24\ 10^{-3}$
3	$-5.17\ 10^{0}$	$7.19\ 10^{-3}$	$9.28\ 10^{-3}$	$-1.44\ 10^{0}$	$5.44\ 10^{-3}$	$2.26\ 10^{-3}$
4	$-5.04\ 10^{0}$	$2.95\ 10^{-1}$	$3.28\ 10^{-2}$	$-1.38\ 10^{0}$	$6.63\ 10^{-2}$	$8.99\ 10^{-3}$
5	$-2.37\ 10^{0}$	$2.52\ 10^{-1}$	$3.37\ 10^{-1}$	$-6.82\ 10^{-1}$	$7.31\ 10^{-2}$	$9.56\ 10^{-2}$
6	$-3.65\ 10^{-1}$	$2.87\ 10^{-1}$	$2.38\ 10^{-1}$	$-1.08\ 10^{-1}$	$7.69\ 10^{-2}$	$6.43\ 10^{-2}$
7	$9.69\ 10^{-3}$	$4.82\ 10^{-2}$	$4.49\ 10^{-2}$	$-7.16\ 10^{-3}$	$1.05\ 10^{-2}$	$9.91\ 10^{-3}$
8	$-4.88\ 10^{-3}$	$1.29\ 10^{-2}$	$1.70\ 10^{-2}$	$6.21\ 10^{-4}$	$8.80\ 10^{-3}$	$1.11\ 10^{-2}$
9	$-8.79\ 10^{-3}$	$1.48\ 10^{-2}$	$1.55\ 10^{-2}$	$4.49\ 10^{-2}$	$3.35\ 10^{-2}$	$3.29\ 10^{-2}$
10	$-2.23\ 10^{-2}$	$2.22\ 10^{-2}$	$1.97\ 10^{-2}$	$4.58\ 10^{-1}$	$1.23\ 10^{-1}$	$1.49\ 10^{-1}$
11	$-9.45\ 10^{-2}$	$2.29\ 10^{-2}$	$2.10\ 10^{-2}$	$5.75\ 10^{0}$	$1.43\ 10^{0}$	$8.95\ 10^{-1}$
12	$-4.53\ 10^{-1}$	$2.41\ 10^{-2}$	$1.79\ 10^{-2}$	$4.47\ 10^{1}$	$4.67\ 10^{0}$	$3.16\ 10^{0}$
13	$-3.77\ 10^{0}$	$1.32\ 10^{-2}$	$8.24\ 10^{-3}$	$1.65\ 10^{1}$	$4.21\ 10^{-1}$	$2.52\ 10^{-1}$
14	$-4.34\ 10^{0}$	$1.20\ 10^{-2}$	$7.20\ 10^{-3}$	$4.01\ 10^{0}$	$2.82\ 10^{-1}$	$1.67\ 10^{-1}$
15	$3.06\ 10^{-1}$	$2.68\ 10^{-2}$	$2.04\ 10^{-2}$	$7.32\ 10^{1}$	$2.44\ 10^{0}$	$2.94\ 10^{0}$

Table C.4. Estimation Results With Both Low and High Frequency Mode Additions

Run	$\widehat{\theta}-\theta$	σ_e	σ_p	$\widehat{\theta}-\theta$	σ_e	σ_p
	$\theta = a1 = -0.84$			$\theta = a2 = -1.44$		
1	$8.59\ 10^{-1}$	$2.14\ 10^{-4}$	$1.73\ 10^{-4}$	$1.44\ 10^{0}$	$2.84\ 10^{-6}$	$3.30\ 10^{-6}$
2	$8.56\ 10^{-1}$	$1.30\ 10^{-4}$	$1.06\ 10^{-4}$	$1.44\ 10^{0}$	$6.08\ 10^{-6}$	$3.22\ 10^{-6}$
3	$8.38\ 10^{-1}$	$3.75\ 10^{-5}$	$1.91\ 10^{-5}$	$1.43\ 10^{0}$	$6.34\ 10^{-7}$	$7.30\ 10^{-7}$
4	$8.34\ 10^{-1}$	$3.45\ 10^{-5}$	$2.99\ 10^{-5}$	$1.43\ 10^{0}$	$1.28\ 10^{-6}$	$1.90\ 10^{-6}$
5	$1.22\ 10^{0}$	$6.41\ 10^{-3}$	$3.09\ 10^{-3}$	$1.45\ 10^{0}$	$1.63\ 10^{-3}$	$5.90\ 10^{-4}$
6	$3.60\ 10^{-1}$	$5.03\ 10^{-2}$	$4.11\ 10^{-2}$	$3.36\ 10^{-1}$	$7.43\ 10^{-2}$	$5.73\ 10^{-2}$
7	$-3.78\ 10^{-3}$	$1.01\ 10^{-2}$	$1.09\ 10^{-2}$	$-7.98\ 10^{-3}$	$7.66\ 10^{-3}$	$7.15\ 10^{-3}$
8	$5.47\ 10^{-4}$	$2.37\ 10^{-3}$	$2.70\ 10^{-3}$	$-1.86\ 10^{-4}$	$2.08\ 10^{-3}$	$2.99\ 10^{-3}$
9	$1.36\ 10^{-3}$	$4.05\ 10^{-3}$	$3.64\ 10^{-3}$	$-7.12\ 10^{-3}$	$5.74\ 10^{-3}$	$6.29\ 10^{-3}$
10	$-5.39\ 10^{-2}$	$2.15\ 10^{-2}$	$2.53\ 10^{-2}$	$-7.18\ 10^{-2}$	$1.35\ 10^{-2}$	$1.59\ 10^{-2}$
11	$-1.04\ 10^{0}$	$2.81\ 10^{-1}$	$1.76\ 10^{-1}$	$-8.38\ 10^{-1}$	$1.14\ 10^{-1}$	$8.86\ 10^{-2}$
12	$-9.08\ 10^{0}$	$9.90\ 10^{-1}$	$6.66\ 10^{-1}$	$-7.39\ 10^{0}$	$5.86\ 10^{-1}$	$3.07\ 10^{-1}$
13	$-5.74\ 10^{0}$	$1.37\ 10^{-1}$	$5.64\ 10^{-2}$	$-8.96\ 10^{1}$	$2.84\ 10^{-1}$	$2.91\ 10^{-1}$
14	$-6.66\ 10^{-1}$	$1.12\ 10^{-1}$	$4.20\ 10^{-2}$	$-2.77\ 10^{2}$	$7.10\ 10^{-1}$	$6.96\ 10^{-1}$
15	$-1.44\ 10^{1}$	$4.03\ 10^{-1}$	$4.47\ 10^{-1}$	$-5.35\ 10^{2}$	$5.11\ 10^{0}$	$4.11\ 10^{0}$
	$\theta = b1 = 4.8$			$\theta = b2 = 1.44$		
1	$-4.80\ 10^{0}$	$2.70\ 10^{-5}$	$2.45\ 10^{-5}$	$-1.44\ 10^{0}$	$1.15\ 10^{-7}$	$1.28\ 10^{-7}$
2	$-4.79\ 10^{0}$	$4.80\ 10^{-5}$	$4.60\ 10^{-5}$	$-1.44\ 10^{0}$	$1.83\ 10^{-6}$	$8.78\ 10^{-7}$
3	$-4.79\ 10^{0}$	$4.71\ 10^{-5}$	$6.93\ 10^{-5}$	$-1.44\ 10^{0}$	$1.26\ 10^{-5}$	$5.83\ 10^{-6}$
4	$-4.80\ 10^{0}$	$8.86\ 10^{-5}$	$1.08\ 10^{-4}$	$-1.44\ 10^{0}$	$2.52\ 10^{-5}$	$1.82\ 10^{-5}$
5	$-5.48\ 10^{0}$	$5.81\ 10^{-3}$	$4.28\ 10^{-3}$	$-1.46\ 10^{0}$	$2.90\ 10^{-3}$	$1.03\ 10^{-3}$
6	$-1.33\ 10^{0}$	$2.85\ 10^{-1}$	$2.22\ 10^{-1}$	$-2.55\ 10^{-1}$	$8.64\ 10^{-2}$	$6.60\ 10^{-2}$
7	$6.07\ 10^{-2}$	$4.96\ 10^{-2}$	$4.60\ 10^{-2}$	$3.10\ 10^{-2}$	$1.10\ 10^{-2}$	$1.03\ 10^{-2}$
8	$-1.79\ 10^{-3}$	$1.29\ 10^{-2}$	$1.70\ 10^{-2}$	$2.25\ 10^{-4}$	$8.79\ 10^{-3}$	$1.11\ 10^{-2}$
9	$-1.99\ 10^{-2}$	$1.47\ 10^{-2}$	$1.55\ 10^{-2}$	$2.85\ 10^{-2}$	$3.34\ 10^{-2}$	$3.28\ 10^{-2}$
10	$-3.74\ 10^{-2}$	$2.22\ 10^{-2}$	$1.97\ 10^{-2}$	$4.30\ 10^{-1}$	$1.23\ 10^{-1}$	$1.48\ 10^{-1}$
11	$-1.10\ 10^{-1}$	$2.28\ 10^{-2}$	$2.10\ 10^{-2}$	$5.70\ 10^{0}$	$1.42\ 10^{0}$	$8.90\ 10^{-1}$
12	$-4.68\ 10^{-1}$	$2.40\ 10^{-2}$	$1.79\ 10^{-2}$	$4.45\ 10^{1}$	$4.66\ 10^{0}$	$3.15\ 10^{0}$
13	$-3.77\ 10^{0}$	$1.31\ 10^{-2}$	$8.21\ 10^{-3}$	$1.64\ 10^{1}$	$4.20\ 10^{-1}$	$2.51\ 10^{-1}$
14	$-4.34\ 10^{0}$	$1.19\ 10^{-2}$	$7.17\ 10^{-3}$	$4.00\ 10^{0}$	$2.81\ 10^{-1}$	$1.67\ 10^{-1}$
15	$2.90\ 10^{-1}$	$2.68\ 10^{-2}$	$2.03\ 10^{-2}$	$7.30\ 10^{1}$	$2.43\ 10^{0}$	$2.93\ 10^{0}$

Table C.5. Estimated Poles and Zeros for Each Case

Run	2nd Order	4th Order Low	4th Order High	6th Order
Poles - True = $-0.42 + j1.124$				
1	0.356 & 0.002	0.009+j0.032	0.353 & 0.002	0.009+j0.032
2	0.356 & 0.004	0.008+j0.053	0.354 & 0.004	0.008+j0.053
3	0.365 & 0.012	-0.001+j0.074	0.362 & 0.012	-0.001+j0.074
4	0.181+j0.152	-0.003+j0.073	0.179+j0.157	-0.003+j0.073
5	-0.181+j0.861	0.414 & -0.031	-0.183+j0.861	0.412 & -0.030
6	-0.382+j1.095	-0.237+j1.022	-0.384+j1.094	-0.240+j1.023
7	-0.421+j1.124	-0.420+j1.128	-0.422+j1.123	-0.422+j1.127
8	-0.419+j1.125	-0.420+j1.124	-0.419+j1.125	-0.420+j1.124
9	-0.419+j1.123	-0.419+j1.122	-0.420+j1.128	-0.419+j1.128
10	-0.416+j1.126	-0.415+j1.125	-0.448+j1.146	-0.447+j1.145
11	-0.399+j1.116	-0.398+j1.116	-0.943+j1.178	-0.941+j1.180
12	-0.435+j1.148	-0.434+j1.148	-8.934 & -0.987	-8.928 & -0.989
13	-0.633+j1.017	-0.634+j1.018	-3.288+j8.954	-3.288+j8.955
14	-0.808+j0.893	-0.809+j0.894	-0.753+j16.669	-0.753+j16.669
15	-0.301+j2.119	-0.301+j2.119	-7.622+j21.881	-7.622+j21.881
Zero - True = 0.300				
1	0.002	0.006	0.002	0.006
2	0.004	0.024	0.004	0.024
3	0.012	0.149	0.012	0.149
4	0.226	5.038	0.236	5.596
5	-0.315	-0.035	-0.312	-0.033
6	-0.302	-0.344	-0.300	-0.341
7	-0.300	-0.305	-0.298	-0.303
8	-0.301	-0.300	-0.300	-0.300
9	-0.296	-0.294	-0.310	-0.307
10	-0.292	-0.287	-0.397	-0.393
11	-0.254	-0.250	-1.528	-1.521
12	-0.331	-0.328	-10.609	-10.602
13	-0.732	-0.732	-17.372	-17.372
14	-1.087	-1.087	-11.838	-11.838
15	-0.082	-0.082	-14.625	-14.624

Appendix D. Results for Chapter V

This appendix contains the normalized numerical results for the higher noise experiments in Chapter V. Tables D.1 and D.2 contain the results for experiments with input noise only, and the others contain the results for the experiments with both input and output noise. Given in each case are the actual average error (e) and error sigma (σ_e), along with the algorithm predicted error sigma (σ_p) for the 100 run Monte-Carlo analysis. The NaN and Inf entries indicate that the particular algorithm did not converge in that case. NaN and Inf are the not-a-number and infinity values, respectively, that are produced by Matlab.

Table D.1. Numerical Results for $p = 3$ - Output Noise Only

Method	θ	$e = \hat{\theta} - \theta$ (%)	σ_e	σ_p	$\hat{\theta} - \theta$ (%)	σ_e	σ_p
		Initial Time			No Initial Time		
Least Squares	a_1	-16.3283	1.3336	0.8827	-15.4486	1.2063	0.8573
	a_2	-30.5571	2.5130	1.6678	-29.0724	2.2751	1.6293
	b_1	-11.8435	1.2978	6.1293	-9.0016	1.2583	6.2259
	b_2	-34.4039	2.7767	6.9777	-29.2986	2.6407	6.9806
Generalized Minimum Variance	a_1	0.0005	0.0381	0.0413	-0.0034	0.0404	0.0415
	a_2	0.0016	0.0825	0.0882	-0.0066	0.0871	0.0885
	b_1	-0.0244	0.3395	0.3443	0.0309	0.3428	0.3444
	b_2	-0.0268	0.3392	0.3470	0.0301	0.3391	0.3467
Generalized Least Squares	a_1	0.0002	0.0380	0.0404	-0.0022	0.0407	0.0417
	a_2	0.0011	0.0819	0.0862	-0.0040	0.0876	0.0891
	b_1	-0.0227	0.3358	0.3431	0.0265	0.3420	0.3422
	b_2	-0.0251	0.3374	0.3461	0.0267	0.3378	0.3444
Instrumental Variable	a_1	-0.0003	0.0422	0.9753	0.0038	0.0454	0.9410
	a_2	-0.0011	0.0855	1.8659	-0.0038	0.0895	1.8151
	b_1	-0.0128	1.1521	6.1058	0.2370	0.8167	6.2204
	b_2	-0.0038	1.1974	6.8403	0.3110	0.9930	6.8564

Table D.2. Numerical Results for $p = 4$ - Output Noise Only

Method	θ	$e = \widehat{\theta} - \theta$ (%)	σ_e	σ_p	$\widehat{\theta} - \theta$ (%)	σ_e	σ_p
		Initial Time			No Initial Time		
Least Squares	a_1	-80.2695	1.8686	2.0634	-79.8178	1.7720	2.0549
	a_2	-133.0600	3.9315	4.0202	-132.8790	3.7436	4.0182
	b_1	-217.6903	32.6668	59.1150	-213.0996	35.6261	60.1482
	b_2	-362.5656	37.2700	64.9588	-350.6757	40.2495	65.5272
Generalized Minimum Variance	a_1	-0.0175	0.4260	0.3848	-0.0771	0.4257	0.3857
	a_2	-0.0171	0.9154	0.8188	-0.1460	0.9103	0.8215
	b_1	0.1072	3.3889	3.3907	0.6370	3.4707	3.3981
	b_2	0.0526	3.3985	3.4191	0.6210	3.4506	3.4228
Generalized Least Squares	a_1	0.0138	0.4245	0.3764	-0.0271	0.4309	0.3850
	a_2	0.0487	0.9086	0.8003	-0.0425	0.9199	0.8199
	b_1	0.0009	3.3775	3.3745	0.4608	3.4922	3.3691
	b_2	-0.0650	3.4030	3.4051	0.4416	3.4689	3.3922
Instrumental Variable	a_1	NaN	NaN	NaN	0.0259	0.4596	9.1528
	a_2	NaN	NaN	NaN	-0.0614	0.9135	17.7292
	b_1	NaN	NaN	NaN	2.4422	8.1945	61.4004
	b_2	NaN	NaN	NaN	3.1748	9.9608	67.2974

Table D.3. Numerical Results for $p = 3$ - Input and Output Noise

Method	θ	$e = \widehat{\theta} - \theta$ (%)	σ_e	σ_p	$\widehat{\theta} - \theta$ (%)	σ_e	σ_p
		Initial Time			No Initial Time		
Least Squares	a_1	-17.2891	1.2256	0.8709	-15.9980	1.1173	0.8495
	a_2	-31.2484	2.3315	1.6623	-29.0062	2.1094	1.6277
	b_1	-43.0761	7.7805	4.7164	-39.4903	7.5683	4.7598
	b_2	-69.7460	8.8178	5.3556	-62.9591	8.5739	5.3315
Generalized Minimum Variance	a_1	0.0015	0.0896	0.0877	-0.0130	0.0899	0.0873
	a_2	-0.0053	0.1941	0.1864	-0.0323	0.1906	0.1857
	b_1	-0.5206	0.6930	0.7201	-0.4320	0.6973	0.7155
	b_2	-0.5089	0.7059	0.7268	-0.4174	0.7041	0.7237
Generalized Least Squares	a_1	0.0328	0.0930	0.0404	0.0111	0.0929	0.0417
	a_2	0.0576	0.2029	0.0860	0.0144	0.1965	0.0891
	b_1	-0.7842	0.7207	0.3419	-0.7059	0.7527	0.3413
	b_2	-0.7752	0.7381	0.3449	-0.6906	0.7611	0.3434
Instrumental Variable	a_1	-7.0318	2.6997	1.2022	-5.4413	2.3233	1.1499
	a_2	-8.7355	3.5016	2.0832	-6.3069	2.8940	2.0245
	b_1	-60.5149	17.1172	5.5997	-55.0791	16.3389	5.5930
	b_2	-79.4733	23.9440	7.0957	-69.8840	22.1776	6.8645

Table D.4. Numerical Results for $p = 4$ - Input and Output Noise

Method	θ	$e = \widehat{\theta} - \theta$ (%)	σ_e	σ_p	$\widehat{\theta} - \theta$ (%)	σ_e	σ_p
		Initial Time			No Initial Time		
Least Squares	a_1	-76.0406	2.0994	1.9559	-75.8311	1.8472	1.9655
	a_2	-135.6182	4.3837	3.9279	-135.1656	3.9999	3.9403
	b_1	-65.5936	10.0666	8.8508	-63.9891	10.2143	8.7801
	b_2	-142.2788	10.7713	9.3633	-138.8498	10.6890	9.2873
Generalized Minimum Variance	a_1	-0.1503	0.5865	0.6404	-0.3038	0.6772	0.6487
	a_2	-0.7400	1.2970	1.3586	-1.0400	1.4508	1.3771
	b_1	-28.3322	4.2703	4.8447	-27.2686	4.7292	4.8908
	b_2	-27.5265	4.2037	4.9049	-26.4482	4.7150	4.9586
Generalized Least Squares	a_1	NaN	NaN	Inf	NaN	NaN	Inf
	a_2	NaN	NaN	Inf	NaN	NaN	Inf
	b_1	NaN	NaN	Inf	NaN	NaN	Inf
	b_2	NaN	NaN	Inf	NaN	NaN	Inf
Instrumental Variable	a_1	NaN	NaN	NaN	NaN	NaN	NaN
	a_2	NaN	NaN	NaN	NaN	NaN	NaN
	b_1	NaN	NaN	NaN	NaN	NaN	NaN
	b_2	NaN	NaN	NaN	NaN	NaN	NaN

Bibliography

1. Apostol, Tom M. *Mathematical Analysis* (2nd Edition). Addison-Wesley Publishing Co., 1974.

2. Blakelock, John H. *Automatic Control of Aircraft and Missiles* (2nd Edition). John Wiley & Sons, Inc, 1991.

3. Bloem, J., et al., "System Identification Competition: Benchmark tests for estimation methods of thermal characteristics of buildings and building components." Joint Research Centre, Institute of System Engineering and Informatics, Ispra Italy. Active Period: July 1, 1994 - December 31, 1994.

4. Brown, James M. *Optimal Inputs for System Identification.* Ph.D. dissertation, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, 1995. (AD-A297483)(DA9541967).

5. Coley, D. A. and J. M. Penman. "Second Order System Identification on the Thermal Response of Real Buildings. Paper II: Recursive Formulation for On-Line Building Energy Management and Control," *Building and Environment*, *27*(3):269–277 (1992).

6. Creamer, Glenn. "Applications of Self-Tuning Control for Spacecraft Tracking: Theory and Experiment." *Proceedings of the 1994 American Control Conference*. Number 2 in 3. 2025–2028. June 1994.

7. D'Azzo, John J. and Constantine H. Houpis. *Linear Control System Analysis and Design* (3rd Edition). McGraw-Hill, 1988.

8. Dem'yanov, V. F. and V. N. Malozemov. *Introduction to Minimax*. Dover Publications, Inc., 1974.

9. Eykhoff, P. and P. C. Parks. "Identification and System Parameter Estimation; where do we stand now?," *Automatica*, *26*(1):3–5 (1990).

10. Gauss, K. G. *Theory of Motion of Heavenly Bodies*. New York: Dover, 1963.

11. Gnedenko, B. V. *The Theory of Probability*. Chelsea Pub. Co., 1962.

12. Goodwin, Graham C. and Robert L. Payne. *Dynamic System Identification: Experiment Design and Data Analysis*. New York: Academic Press, 1977.

13. Houpis, Constantine H. and Gary B. Lamont. *Digital Control Systems: Theory, Hardware, Software* (2nd Edition). McGraw-Hill, 1992.

14. Hsia, T. J. *System Identification*. Massachusetts: Lexington Books, 1977.

15. Ingham, Edward A. *Parameter Estimation for Superimposed Weighted Exponentials*. Ph.D. dissertation, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, 1996.

16. Istrăţescu, Vasile I. *Fixed Point Theory*. D. Reidel Publishing Co., 1981.

17. Kharitonov, V. L. "Asymptotic Stability of an Equilibrium Position of a Family of Systems of Linear Differential Equations," *Differential Uravnen.*, *14*:2086–2088 (1978).

18. Klein, Vladislav. "Application of System Identification to High Performance Aircraft." *Proceedings of the 32nd Conference on Decision and Control*. Number 3 in 4. 2253–2258. December 1993.

19. Lin, P. L. and Y. C. Wu. "Identification of Multi-Input Multi-Output Linear Systems From Frequency Response Data," *Trans. ASME, Journal of Dynamic Systems, Measurement, and Control*, *104*:58–64 (March 1982).

20. Ljung, Lennart. *System Identification: Theory for the User*. New Jersey: Prentice-Hall, Inc., 1987.

21. Ljung, Lennart and Svante Gunnarsson. "Adaptation and Tracking in System Identification–A Survey," *Automatica*, *26*(1):7–21 (1990).

22. "Matlab." The MathWorks, Inc., 24 Prime Park Way, Natick, MA 01760, December 1992. Version 4.2b, Copyright 1984-94.

23. Niedzwiecki, Maciej and Lei Guo. "Nonasymptotic Results for Finite-Memory WLS Filters," *IEEE Transactions on Automatic Control*, *36*(2):198–206 (February 1991).

24. Niu, Shaohua and D. Grant Fisher. "A Multiple Model Least-Squares Estimation Method." *Proceedings of the 1994 American Control Conference*. Number 2 in 3. 2231–2235. June 1994.

25. Pachter, M. and P. R. Chandler. "Universal Linearization Concept for Extended Kalman Filters," *IEEE Transactions on Aerospace and Electronic Systems*, *29*(3):946–961 (July 1993).

26. Pearce, B. F., et al. *Analytical Study of Approximate Longitudinal Transfer Functions for a Flexible Airframe*. Technical Report ASD-TDR-62-279, Flight Control Laboratory, Wright-Patterson AFB, OH 45433-6553, June 1962.

27. Pearson, A. E. and Y. Shen. "Weighted Least Squares/MFT Algorithms for Linear Differential System Identification." *Proceedings of the 32nd Conference on Decision and Control*. Number 3 in 4. 2032–2037. December 1993.

28. Phillips, Charles D. and Royce D. Harbor. *Feedback Control Systems*. Prentice Hall, 1988.

29. Pierre, D. A. "On the Simultaneous Identification of Transfer Functions and Initial Conditions." *Proceedings of the 1992 American Control Conference*. Number 2. 1270–1274. June 1992.

30. Pintelon, R., et al. "Parametric Identification of Transfer Functions in the Frequency Domain – A Survey," *IEEE Transactions on Automatic Control*, *39*(11):2245–2260 (November 1994).

31. Ramambason, O. C., et al. "On Using the Parameter Covariance for Improving the Recursive Least-Squares Algorithm." *Proceedings of the 1992 American Control Conference*. Number 2. 1431–1435. June 1992.

32. Steiglitz, K. and L. E. McBride. "A Technique for Identification of Linear Systems," *IEEE Transactions on Automatic Control*, *Ac-10*:461–464 (October 1965).

33. Tektronix Product Catalog, 1993.

34. Unbehauen, H. and G. P. Rao. "Continuous-time Approaches to System Identification–A Survey," *Automatica*, *26*(1):3–5 (1990).

35. Zhou, Quan-Gen and William R. Cluett. "Recusive Identification of Time-Varying Systems via Incremental Estimation." *Proceedings of the 1994 American Control Conference*. Number 1 in 3. 122–126. June 1994.

Vita

Odell R. Reynolds [redacted]. He grew up in Russell but graduated from Marshall Senior High School in 1987. From there he attended North Dakota State University, graduating with honors with a BS in Electrical and Electronics Engineering in May 1991. He received his commission through the Reserve Officer Training Corps, Det 610, and reported for active duty in May 1992. His first assignment was to the Air Force Institute of Technology. He graduated from there with his Masters of Science in Electrical Engineering in December 1993. He pursued his PhD, also at the Air Force Institute of Technology, from January 1994 to December 1996. He married Stacey Olson in [redacted] His career is scheduled to continue in the Air Force at Wright Laboratory.

[redacted]

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
December 1996

3. REPORT TYPE AND DATES COVERED
Dissertation

4. TITLE AND SUBTITLE
COUNTERING THE EFFECTS OF MEASUREMENT NOISE DURING THE IDENTIFICATION OF DYNAMICAL SYSTEMS

5. FUNDING NUMBERS

6. AUTHOR(S)
Odell R. Reynolds
Captain, USAF

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology, WPAFB OH 45433-6583

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/DS/ENG/96-13

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Lt Christina L. Osmon
WL/FIGS
2210 Eighth St STE 11
WPAFB, OH 45433-7521

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; Distribution Unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Sensor noise is an unavoidable fact of life when it comes to measurements on physical systems, as is the case in feedback control. Therefore, it must be properly addressed during dynamic system identification. In this work, a novel approach is developed toward the treatment of measurement noise in dynamical systems. This approach hinges on proper stochastic modeling, and it can be adapted easily to many different scenarios, where it yields consistently good parameter estimates. The Generalized Minimum Variance algorithm developed and used in this work is based on the theory behind the minimum variance identification process, and the estimate produced is a fixed point of a mapping based on the minimum variance solution. Additionally, the algorithm yields an accurate prediction of the estimation error. This algorithm is applied to many different noise models associated with three basic identification problems. First, continuous-time systems are identified using frequency domain measurements. Next, a discrete-time plant is identified using discrete-time measurements. Finally, the physical parameters of a continuous-time plant are identified using sampled measurements of the continuous-time input and output. Validation of the estimates is performed correctly, and the results are compared with other, more common, identification algorithms. The GMV results are generally better.

14. SUBJECT TERMS
System Identification, Measurement Noise, Dynamical Systems, Generalized Minimum Variance

15. NUMBER OF PAGES
153

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT
UNCLASSIFIED

18. SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

19. SECURITY CLASSIFICATION OF ABSTRACT
UNCLASSIFIED

20. LIMITATION OF ABSTRACT
UL

GENERAL INSTRUCTIONS FOR COMPLETING SF 298

The Report Documentation Page (RDP) is used in announcing and cataloging reports. It is important that this information be consistent with the rest of the report, particularly the cover and title page. Instructions for filling in each block of the form follow. It is important to ***stay within the lines*** to meet ***optical scanning requirements***.

Block 1. Agency Use Only *(Leave blank)*.

Block 2. Report Date. Full publication date including day, month, and year, if available (e.g. 1 Jan 88). Must cite at least the year.

Block 3. Type of Report and Dates Covered. State whether report is interim, final, etc. If applicable, enter inclusive report dates (e.g. 10 Jun 87 - 30 Jun 88).

Block 4. Title and Subtitle. A title is taken from the part of the report that provides the most meaningful and complete information. When a report is prepared in more than one volume, repeat the primary title, add volume number, and include subtitle for the specific volume. On classified documents enter the title classification in parentheses.

Block 5. Funding Numbers. To include contract and grant numbers; may include program element number(s), project number(s), task number(s), and work unit number(s). Use the following labels:

C	- Contract	PR	- Project
G	- Grant	TA	- Task
PE	- Program Element	WU	- Work Unit Accession No.

Block 6. Author(s). Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. If editor or compiler, this should follow the name(s).

Block 7. Performing Organization Name(s) and Address(es). Self-explanatory.

Block 8. Performing Organization Report Number. Enter the unique alphanumeric report number(s) assigned by the organization performing the report.

Block 9. Sponsoring/Monitoring Agency Name(s) and Address(es). Self-explanatory.

Block 10. Sponsoring/Monitoring Agency Report Number. *(If known)*

Block 11. Supplementary Notes. Enter information not included elsewhere such as: Prepared in cooperation with...; Trans. of...; To be published in.... When a report is revised, include a statement whether the new report supersedes or supplements the older report.

Block 12a. Distribution/Availability Statement. Denotes public availability or limitations. Cite any availability to the public. Enter additional limitations or special markings in all capitals (e.g. NOFORN, REL, ITAR).

DOD	- See DoDD 5230.24, "Distribution Statements on Technical Documents."
DOE	- See authorities.
NASA	- See Handbook NHB 2200.2.
NTIS	- Leave blank.

Block 12b. Distribution Code.

DOD	- Leave blank.
DOE	- Enter DOE distribution categories from the Standard Distribution for Unclassified Scientific and Technical Reports.
NASA	- Leave blank.
NTIS	- Leave blank.

Block 13. Abstract. Include a brief *(Maximum 200 words)* factual summary of the most significant information contained in the report.

Block 14. Subject Terms. Keywords or phrases identifying major subjects in the report.

Block 15. Number of Pages. Enter the total number of pages.

Block 16. Price Code. Enter appropriate price code *(NTIS only)*.

Blocks 17. - 19. Security Classifications. Self-explanatory. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED). If form contains classified information, stamp classification on the top and bottom of the page.

Block 20. Limitation of Abstract. This block must be completed to assign a limitation to the abstract. Enter either UL (unlimited) or SAR (same as report). An entry in this block is necessary if the abstract is to be limited. If blank, the abstract is assumed to be unlimited.

★U.S.GPO:1990-0-273-271